Peer Production of Open Hardware: Unfinished Artefacts and Architectures in the Hackerspaces

Information and Knowledge Society Doctoral Programme

Universitat Oberta de Catalunya (UOC)

Author: Péter Dunajcsik a.k.a. Maxigas

Research group: Open Science and Innovation

Supervisor: Eduard Aibar (Universitat Oberta de Catalunya)

Committee members:

- Andrew Pickering (University of Exeter)
- Johan Söderberg (Göteborgs Universitet)

. Barcelona, June 8, 2015 .

Contents

Bill of Parts

1 Introduction

1.1 Foreword

It is daunting to spend three years writing a thesis which lives in a folder called "unfinished". Despite the bad omens, however, the project rolled fine and came to a conclusion just before its designated deadlines. One reason is that is it started well before the doctoral programme with a Masters' thesis in 2011 entitled *This is r0ket science! Modernity, Capitalism and Liberalism in Hacker Culture* – which featured a similar research question revolving around peer production, as well as the first case study included here. At that time I was mainly trying to exploit my position as a co-founder of a hackerspace in Budapest and earlier a hacklab in London to produce a critique of the scene which is readable for both hackers and academics. However, the experiences which went into the thesis are even older, dating back more than a decade, when I first went to university, became an anarchist, a media activist, and later a squatter. It was only as a doctoral candidate that I managed to combine my diverse interests in technology, politics and self-organisation coherently and comfortably.

1.2 Research questions

1.2.1 Peer production of open source hardware

The overall attempt of the dissertation is to give an account of *peer production* as a set of social practices, rather than an economic, legal or organisation regime. Moreover, to provide a critique of current collaborative practices and their theoretical formulations – a critique which understands these phenomena as an integral yet potentially subversive part of the history of modernity, liberalism and capitalism, rather than as a new or emergent phenomena. There are a number of texts which do similar things, but these academic discussions are sometimes dominated too much by visionary fantasies rather than the close reading of the present conditions. Therefore, I wanted a text which presents its theoretical assertions backed up eminently by detailed ethnographic data, rather than speculations.

The main research question is **how the model of *commons-based peer production* is transformed when transplanted to a different domain, namely from open source software to *open source hardware*?** I study this question in the context of the hackerspaces, through two cases of small scale electronic artefacts. I argue that the hackerspaces are a potentially paradigmatic research site because hackers are widely pictured as eminent practitioners of commons-based peer production and the hackerspaces are where they produce open source hardware collectively. Since the free, libre and open source software development model associated with GNU/Linux is the principal example in the literature for commons-based peer production, I contrast my findings with

FLOSS (Free, Libre and Open Source Software) practices. Major studies of hacker culture (Himanen 2001; Söderberg 2008; Weber 2004) identify the question *how commons-based peer production works in the physical realm*, in particular on hardware, as a promising area for future research. I have recently edited (with Peter Troxler) the *Journal of Peer Production* Issue 5 on "Shared Machine Shops", which investigate how peer production practices work when embodied communities make open source hardware collaboratively in shared laboratories.

1.2.2 Subquestions: participation and expertise

In order to narrow down the scope of the investigation, I pay particular attention to two interlocking themes, asking **how *participation* and *expertise* are co-articulated in the hackerspaces during the peer production of open source hardware projects?** There is obviously a tension between the full inclusiveness (open door policy) of their grassroots spaces and the highly technological content of the hackers' work, which have to be negotiated through specific social practices, technologies and forms of organisation. In this context, I want to *understand the role of expertise in the organisation and function of hackerspaces and the way the particular forms of expertise developed in the hackerspaces are expressed and solidify as technological choices in the artefacts which are created and shared.* As STS scholar Andrew Feenberg notes, participation in the production and deployment of technology is key to understanding political processes: "technology is one of the major sources of public power in modern societies. So far as decisions affecting our daily lives are concerned, political democracy is largely overshadowed by the enormous power wielded by the masters of technical systems" (1992, 301). I plan to use of Feenberg's theoretical framework on participation as one of the starting points of my investigation. I ask *how participation works in these electronic projects, how commons-based peer production is implemented in the case of physical devices and what difference participation and peer production makes in the actual technical architectures which result from such efforts.*

1.2.3 Crosscutting concerns

Crosscutting concerns further orient the investigation, delineating aspects of special interest that can be picked up during the analysis of the ethnographic material.

1.2.3.1 What are the political contexts and consequences of such and such arrangements? That is, whether peer production can be disruptive to capitalism or is it merely transformative, taking it to a new level? How the political economy of hardware production differs from software production? Is participation as it happens in the hackerspaces empowering or contributing to a democratisation process in any way? How the social construction of expertise in the hacker scene changing who can have effective input into policy processes?

These questions are considered in the light of Söderberg's observation that hackers have a tendency to pose political questions as technical problems – a usual technocratic manoeuvre [-Soderberg2013a]. However, at the same time, hackers often do politicise areas or elements that were seen as neutral questions of technical or market efficiency before. Therefore, the classic STS question of how and why certain things are framed as political and certain things remain technical – that is, politically neutral - should be the central concern here.

1.2.3.2 What are the technological consequences of such and such arrangements? That is, what actual difference openness makes in the technical solutions and their technical implementation, and finally, in the functioning of the resulting products? How is a community produced technological artefact different from a mass produced artefact in terms of the technological choices that have been made, and inversely: what kind of technological choices are encouraged or made more difficult by the hackerspace model of participation? How concepts of expertise are reflected, and in turn made possible, by the particular architectures of open hardware in contrast to mass products available on the market?

These questions are considered in the light of the actual technical details of artefacts, together with their embeddedness in specific social practices. Where technology is questioned the different solutions for solving the same problem and the different problems that could have been solved has to be considered carefully, because technological choice manifests itself through the grammar of these choices. Technology is therefore not understood in the sense of technological determinism – having its own autonomous teleology and logic –, but as a social construction full of contingencies.

1.3 Structure of the dissertation

The general trajectory of the text is as follows. After giving the reader just enough explanations to make sense of the project as a whole, as well as the most obscure ideas involved, the theoretical framework offers a discussion of key concepts such as *peer production*, *open source hardware*, *participation* and *expertise*. It can be understood as a rather essayistic literature review. The methods chapter follows, laying out how the research project was made and giving extensive justifications for making it that way. The free software chapter connects these introductory chapters to the main body of the thesis. It is a liminal chapter which concentrates on the recapitulation of the state of knowledge on free software, which serves as the background against which it is possible to see the idiosyncrasies of open hardware later on. Owing to its position, – even though it contains many critical remarks and much original research – it is still mainly parasitic on previous publications. I have to explain *my* understanding of free software, which is not based on properly presented ethnographic data, in order to render intelligible subsequent findings on open source hardware.

The body proper is formed by five empirical chapters: the first three construct the research site where the ethnographic work took place, yet it is also a pocket monograph of the hackerspaces scene. The second two is where the real actors, the technological artefacts which are the basis of the case studies really enter the stage. The two case studies are explicitly *not* tied into in a systematic comparative framework, but presented in a sequential order so that a consistent argument about my themes can be presented in subsequent stages. The case studies are where the concepts in the title – *unfinished artefacts and architectures* – are fleshed out gradually. In particular, the first case study concentrates on the reinterpretation of open source hardware as unfinished artefacts, while the second case study concentrates on the reinterpretation of hackerspaces as unfinished architectures. I introduced these concepts to reformulate both the peer production and the open source hardware concepts, to transform them using my contributions based on ethnographically grounded findings. The differences between past readings of peer production / open hardware and unfinished architectures / unfinished artefacts are more explicitly formulated, gathered and distilled in the *Conclusions* chapter. Finally, the *Afterword* identifies the possibilities for the continuation of the research project, as well as its consequences to broader historical and intellectual questions.

1.4 Outline of research design

The project is structured in the following way:

- Question: *implementation of peer production in hardware.*
- Subquestions: *role of participation and expertise.*
- Methodological framework: **peer production**, **material semiotics**, **critical theory of technology**.
- Research sites: **hackerspaces** in the Netherlands, Germany and Eastern Europe.
- Case studies: **open source hardware projects**, namely
    1. the r0ket device; and
    2. space state systems.
- Methods:
    - *object biographies*,
    - *critical historiography*,
    - *technical interrogation.*
- Feasibility: ongoing research with native access to the field.

Figure 1. provides a visual summary of how the overall research theme, the aspects of the research, and the crosscutting concerns fit together.

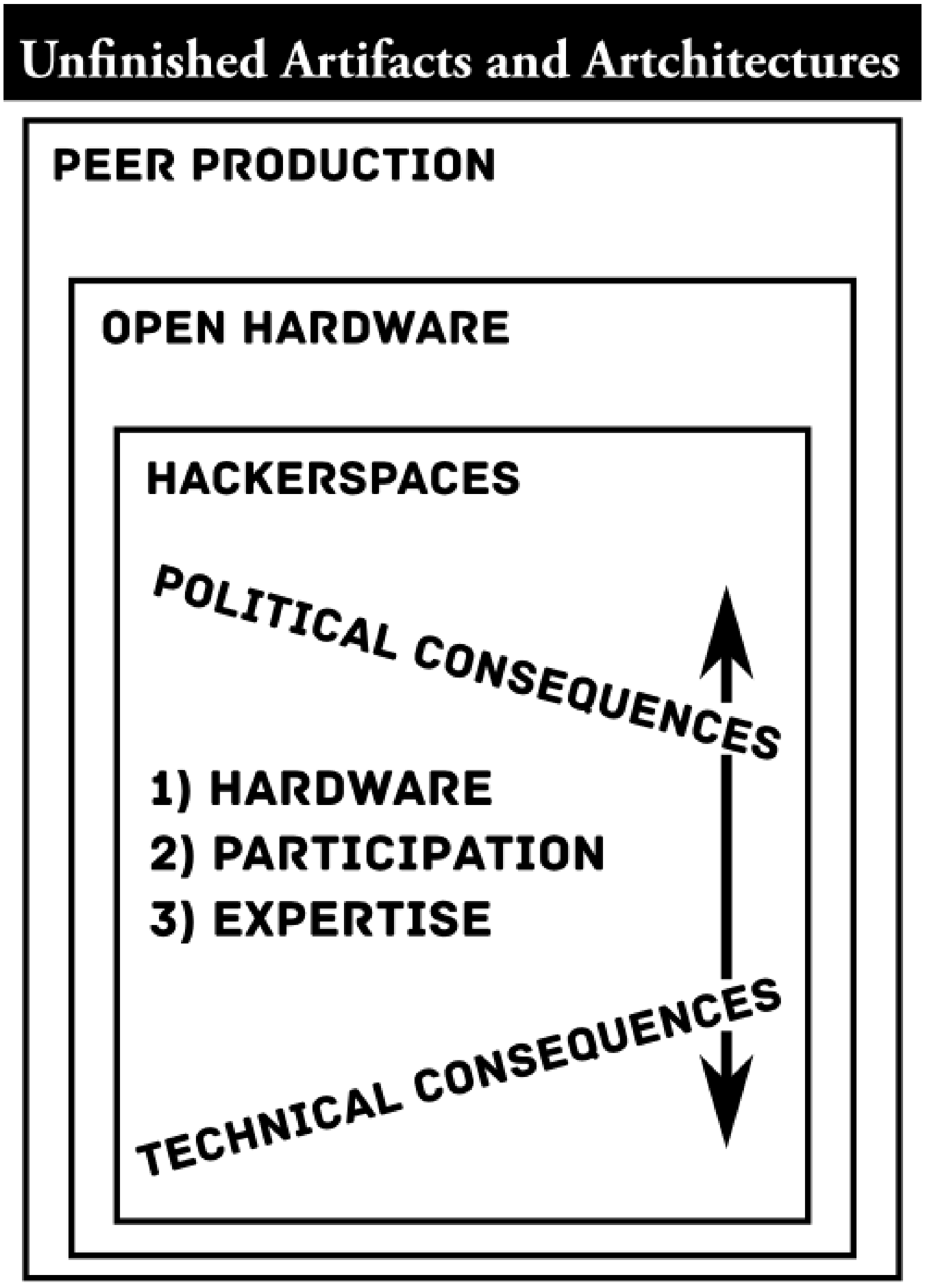


Figure 1: Diagram of the research plan.

2 Definition of key concepts

2.1 Theoretical framework: Peer production

Peer production is a form of network-based voluntary cooperation aimed at contributing to a commons, epitomised by the Linux kernel and Wikipedia and more recently applied to hardware (Benkler 2006; Boyle 2013; Wainwright and Bauwens 2012; Söderberg 2011). Digital and material commons are essential elements of peer production ecosystems both as raw material, organisational enablers and end products. Peer production is theorised as a mode of production or a form of organising labour which gained prominence and character through social transformations in post-industrial, network or knowledge societies (Franco Berardi a.k.a. Bifo 2005; Castells 1996; Castells 1997; Castells 1998). As such, the functioning of material peer production on the level of social practices is not well understood.

At the same time, its structural consequences regarding the transformation of life, labour and society are found to be overwhelming. Interpretations regarding the latter range from disruptive changes in liberal democratic market capitalism (Benkler 2006), visions of social democratic flexicurity (Wilthagen and Tros 2004) or deepening exploitation based on free labour (Terranova 2000), and even emerging communism (Rigi 2013). *Evidently, more research is required to clarify the issue, especially one which confronts these diverse perspectives and where empirical evidence plays a key role in the presentation of arguments.*

2.2 Research sites: Hackerspaces

Hackerspaces are paradigmatic research sites for the study of peer production of open hardware. The hacker scene in general, which is internally diverse (Coleman 2012), and in particular the hackerspaces scene (Altman 2012a; Pettis 2008; esp. Bre and Astera 2008), can be considered the homeland of commons based peer production practices where they can be studied in situ. Once, I argue it achieved a relative autonomy from market pressure, state control and even the institutions of civil society (Maxigas 2012a), which provides a fertile ground for experimentation with alternative forms of production and organisation of work. Twice, hackerspaces are explicitly set up with the single mission to facilitate commons based peer production of open hardware, in the context of the social milieu which nurtures it. Thrice, hackerspaces are embodied communities where peer production practices can be studied in the context of everyday life combining the most advantageous aspects of well-established traditional ethnographic methods and digital methods which look at online platforms on which community interactions are played out.

So what are hackerspaces? **Hackerspaces provide the opportunity for technological practitioners to meet, socialise and work on projects**

in a well equipped collaborative environment outside of professional contexts. They are a generation of shared machine workshops or urban labs which proliferated mainly since 2005 (Maxigas 2012a) and now number over two thousand, mainly located in the cities of Europe and North America. Membership numbers of individual hackerspaces range from few dozen to several hundred.

They are hotbeds of grassroots research and development (sometimes called open innovation), self-organised political resistance (mainly in the area of techno-politics and info-politics), collaborative and constructivist autodidactism (informal education) and self-managed production (craftsmanship). In this capacity they are very interesting sites where three functions traditionally separated in modern institutions blend: (a.) education; (b.) development; and (c.) production.

Formally, hackerspaces usually assume the form of associations or foundations, and cover their operating costs through membership fees – in this sense they are like clubs, providing a socialising and working environment for their members. However, when any member "opens" the hackerspace, it is usually automatically open to the general public. Anybody can go and use tools like 3D printers, laser cutters, welding machines or electronics and biology laboratory equipment, etc. as well as receive advice from knowledgeable members. In this capacity they provide the physical infrastructure and tutoring capacity which complements the free software and documentation available online. Open hardware needs open laboratories which provide the tools and the communities of practice that are necessary for the development of electronics. Hackerspaces are the sites for the education of non-professional technologists often with a professional level of expertise.

2.3 Case studies: Open source hardware projects

Open source hardware projects are crucial for the understanding of peer production for a number of reasons. Firstly, **most of the existing literature is based on studies of free software development** (typically, Linux) or knowledge production (typically, Wikipedia). Therefore, it is important to extend the literature to see how peer production works in other fields (Paoli and Storni 2011).

Secondly, on the one hand, open hardware projects (Association 2008) are an **interesting case for inquiring into the general applicability of peer production** – on which many of the more daring conclusions rest –, and on the other hand, they are but *a half step away from immaterial labour* in the sense that software and documentation are still an essential part of the functioning of open hardware projects (Ackermann 2009; CERN 2012).

Thirdly, there is **a very strong bias in the literature towards studying the most successful projects in the most successful moments of their trajectory**, which leads straightforwardly to theoretical conclusions that peer production projects are both incredibly successful and extremely disruptive. I propose to counterbalance such bias by studying media scale open hardware projects which are closer to the norm in hackerspaces through the whole of their trajectory, from inception to obsolescence.

Ultimately, the question of hardware puts into play **issues of materiality** which are at the centre of current theoretical discussions in the social sciences, especially Science and Technology Studies.

My research project is based on the ***object biography*** **of two technological artefacts**: the r0ket badge, and door systems:

1. The *r0ket badge* is a mobile phone alternative which creates a wireless mesh network. Gathering the cultural triggers of the hackerspaces scene, it was distributed in the most important hacker gatherings in Europe (with two to four thousand participants). It has been developed in µC3 hackerspace in München, Germany.

2. *Door systems* are Do It Yourself infrastructures for signalling and reporting when a hackerspace is open. A staple of hackerspace, they are used to subvert the social institution of the "opening time". Significant development and deployment exist in Dutch hackerspaces, which I am studying at 12 hackerspaces (listed on hackerspaces.nl).

3 Theoretical framework

The theoretical framework is comprised of two sections. One outlines a baseline theoretical understanding of technology that informs my approach to the research questions. The other is delivered in three parts: the discussion of key concepts including peer production, open source hardware and the relationship between expertise and participation. The methods chapter which follows the theoretical framework shows how the concepts developed here are instrumentalised in the research strategy and practice.

3.1 Concepts of technology

My overall understanding of technology is informed by the *critical theory of technology*, along the lines of Feenberg (2002) and Kirkpatrick (2008). **The critical theory of technology analyses technology as a form of human endeavour and material culture which – as all other spheres of human activity – is deeply embedded in social history (Ibid. 5-10).** Therefore, technology is *socially constructed*: what technology is (its *essence*) and how it is applied to specific problems (its *use*) are worked out by people bound in the social conflicts of their day. Furthermore, technology is a *material practice*, so its functioning and development is closely tied to the negotiations with matter, with known and unknown physics and mathematics, and most importantly with the current state of the art. These two complementary aspects, social and material, are called *secondary and primary instrumentalisation*, respectively (Feenberg 2005).

In broad terms, **the specificity of our particular historical era is grasped through the concepts of modernity, capitalism and liberalism**. More narrowly, the contemporary moment stretching from the 1970s to now is theorised as *reflexive modernity* (Beck, Bons, and Lau 2003; Beck, Giddens, and Lash 1994), *late capitalism* (Jameson 1991) and the time of *neoliberal hegemony* (Harvey 2005). Finally, the most recent structural changes in production – which are essential to keep in mind for any discussion of peer production – have been theorised as the *project order* (Boltanski and Chiapello 2005), the *network society* (Castells 1996; Castells 1997; Castells 1998) and lately as *cognitive capitalism* (Boutang 2011). I show in the case studies how hackers negotiate the social contradictions produced by these transformations through socio-technical inventions like the r0ket device. This hand held electronic artefact can be seen as the rearticulation of late *modernity's individualisation processes* analysed by Giddens (1991), Beck (1992) and Castells (2009, 116–136; 2012, 230; 2014): a *mass self-communication device* like the mobile phone but open in its design and decidedly local in its functionality, technically peer to peer and socially peer produced. The significance of the its unique characteristics can only be appreciated if we take into account the long term trajectory of social history and its contradictions.

Technological orders correspond to these historical periods, even though old technologies often remain in use and become part of a sedimentation of technological layers [Maxigas2014e]. **All in all, what I contend here is nothing more than that what happens always happens against the backdrop of a particular historical horizon, embedded in a geographically and temporarily dispersed context.** What is possible here and now is determined by what happened before in the world.

Of course that doesn't mean "no surprises" – in fact historical contingencies can be recognised for what they are only against the background of overall historical tendencies. While critical theory of technology does not argue for social determinism, it recognises the extra work actors have to put in to go against prevailing trends. Later on I develop the concept of architectures to refer to such effects. Moreover, I use Feenberg's *subversive rationalisation* thesis (1992) to argue how positive socio-technological change can happen. Having said that, while the critical theory of technology is good in situating phenomena in the big picture, it is not ideally equipped to deal with local configurations and to understand the micro-processes at work in the design and deployment of technologies.

As far as the trajectory of particular technologies go – which is the subject of the object biography method expanded in the next chapter –, the theory of the *social construction of technologies* (SCOT) can conceptualise them in terms of *stabilisation* (Bijker and Law 1992). In the early phase of the technology various social groups give different interpretations, called *technological frames*, which are constrained by the *interpretative flexibility* of the given technology. *Closure* sets in if and when the differences between the interpretations are settled. In sum, **stabilisation means that the interpretative flexibility of a technology decreases over time** (Pinch and Bijker 1984).

Recent updates to SCOT (such as contributions in Oudshoorn and Pinch 2003) emphasise the role of *users* in the shaping of technology – a topic central to open hardware which I will pick up later on. Once again, my findings point to the fluidity of technologies, yet also the affective and practical work which goes into maintaining that fluidity against reification. **I develop the concepts of unfinished artefacts and architectures to theorise these problematics, for instance by showing how functional parts in the r0ket badge serve only to forestall stabilisation, offering the possibility of finishing the artefact to its users.**

On the other hand, advances in *material semiotics*[1] in recent decades managed to move issues of *local emergence*, *material agency* and *assemblages* to the centre of discussions (Latour 2005). Authors such as Latour, Law and Callon asked how agency is distributed across networks of human and non-human actors, and how phenomena is constructed locally through enrolling participants and stabilising networks (Law 1991). The focus on localised structures prompted

[1] Often discussed under the rubric of *Actor Network Theory*.

the exploration of *interfaces* and *translations*, since researchers confronted the problematics of mediation between localities. I find such work valuable especially for the treatment of case studies, since it provides a vocabulary in which to express social and technical relations, and situating technological artefacts and human agents in a level playing field within the account (Akrich and Latour 1992). For instance, the functionality of door systems depend on the distribution of agency between human and nonhuman agents, which is clearly seen when tasks such as closing the windows before leaving the room are automated. However, **the challenge is exactly to examine how processes on various scales – from eras through the lifespan of concrete technologies or institutions to the situations between individual actors – interface with one another**. More precisely, how local configurations absorb or resist, create or drive larger scale processes. For ultimately, the existence of large scale processes rests on local affirmation (Latour 1991, 118).

3.2 Discussion of key concepts

3.2.1 Peer production

This section is an attempt to place the theoretical framework of peer production into the context of the Science and Technology Studies research programme: in other words, an STS take on peer production. As evident from preceding discussions, my main interest is the interaction between the (social, technical, etc.) architectures of peer production and the artefacts which result. Therefore my discussion of peer production is oriented towards exploring the role of materiality in the production process and the difference in the products it yields. These particular interests require some limitations and some extensions of the general overview of the state of the art of peer production theories.

3.2.1.1 Extensions A social constructivist look at peer production has to take note of the very history of the theory as well as the actual praxis associated with the concept. In other words, the discussion should give a reflexive account of the theories under consideration. Therefore, **peer production figures here as a concept, a theoretical framework, but at the same time an *ideology* and *political programme*.** Therefore, particular attention is devoted to its critiques. A *relevant question asked is what knowledges participate in the legitimation, justification and organisation of peer production and how they interact?* Such a question is aligned with the subquestion of the thesis about how expertise, in conjunction with participation, is articulated and configured in peer production.

Moreover, if peer production is treated as a material cultural practice, it is through a historically informed close reading of the *laboratory life* that it covers. As such, peer production should not be treated as an abstract formula which gives us a piece of software, a tome of an encyclopedia, or a roadster, but an empirically observable and historically as well as geographically specific socio-technical process which *transforms* its objects – let them be logical functions, knowledge bases or mechanical contraptions – so that its significance is not in whether it can produce this or that but *what kind* of this or that is produced and to *what effect*? Again, the particular question I ask is which social groups participate in peer production and how such participation is organised?

Lastly, if peer production is a socio-technical mechanism – an architecture – oriented towards production, what is the trajectory of these products in terms of the social construction of technology? **How peer production organises the articulation of technological frames, how it reflects or restricts technological flexibility and what courses it offers for stabilisation, what opportunities for closure?** Who decides what works and what not, and on which grounds?

3.2.1.2 Limitations The limitations largely follow from the above. Since I am interested in actual practices in ethnographic detail, I cannot discuss at length claims about "how peer production changes everything" or its role in the "transformation of the social life in the 21st century" since I have not the tools at hand to verify them. **While much of the existing scholarly attention to the topic questions the potentials of peer production for the future of business or for the revolution, my objective is to produce knowledge on the realities of peer production.** Such social prophesies therefore only feature as elements in the ideological discourse of peer production. Instead of offering visionary leadership for the future, my aim is to contextualise these efforts in the social conflicts of the past. *I believe that such an efforts serve the development of scientific knowledge as much as subversive socialities much more, while leaving less opportunities for recuperation.*

Another methodological restriction is that I have no way to verify claims about what is going on inside individuals' heads – therefore, arguments about the psychology of participation in peer production will be largely disregarded. Basically, as far as these do not manifest themselves as practices or at least discourses, they do not count. While interviews and surveys of participants on these topics can provide a useful background to ethnographic findings, I approach them with the hermeneutics of doubt since they do not necessarily reflect the practices of everyday life and likely to reproduce ideological discourse and social imaginaries. The same applies to *ethics*, *motivations* and *desires* – popular topics in the peer production literature.

Finally, **I restrict the discussion to peer production proper, as opposed to hybrid models which combine business logic with open collaboration**. However, my focus is not on the purity of the input – an approach explored by Benkler in his formulation of commons-based peer production. I rather concentrate on the differences in the output: peer-to-peer production instead of peer-to-firm or peer-to-small-time-entrepreneur.

With those qualifications in place, here is an overview of the theoretical discussions around peer production. After a brief introduction reviewing three seminal works that shaped the understanding of peer production phenomena by social scientists, I present the Coasian economic arguments around its efficiency which legitimate the advocacy of peer production by practitioners and theoreticians alike. Where possible, the specific issues on the generality of peer production and the question of its application to tangible artefacts is relegated to the next section on open source hardware.

3.2.1.3 Some central accounts of peer production I singled out three authors whose contribution to the theory of peer production is indispensable. I couple their arguments with the audiences they address and the empirical cases they discuss. I believe these three perspectives are enough to triangulate a "mainstream" or "baseline" of peer production discourse on which subsequent presentations can build. I close the section with addressing the deficiencies in their approach and directions in which I seek to develop what they had to say, using the contributions of STS to the study of technological phenomena in society.

Clay Shirky is probably the one who popularised the theory of peer production most successfully amongst the general public and the general academic audience in *Here Comes Everybody* (2008) – without bringing the term itself to the centre of the debate (probably because he did not come up with it). His *mass amateurisation* thesis made through the case of the blogosphere explores how hobbies and pet projects can become industrially relevant, enabled by the level playing field the Internet provides for information producers and the chaotic processes which organise attention in communication networks with a more or less horizontal topology. Thus the phenomena at hand becomes worthy of scholarly attention. Next, the main argument of the book is the McLuhanite one that changes in ICTs ("media") change social structures, particularly group formations: "When we change the way we communicate, we change society" (17). According to Shirky, everything that changes how groups function have profound ramifications. The latest wave of web platforms enable *new kinds of group-forming* (an idea borrowed from Seb Paquet). Putting the two – mass amateurisation and new group-forming – together, the author refers to such socio-technical systems using Tim O'Reilly's phrase for Web 2.0 platforms, "architectures of participation" (17).

This cuts in closely with my own interest in peer production as a socio-technical architecture which structures participation, which in turn reconfigures expertise and produces technological artefacts with distinctive qualities. The double interest of peer production theory in the role of participation (group-formation) and expertise (amateurisation) in the ambiguously situated double fields of industry and democratisation is evident here. Such double vision in the viewfinder is one of the paradoxes of both peer production theory and praxis with which scholars and practitioners grapple on a page-to-page, day-to-day basis.

His unquestionable lucidity notwithstanding, Shirky writes in the tradition of American bestsellers where any topic addressed by popular science authors have to lead to a "revolution" of some kind, profoundly changing everyday life along with macroeconomic logic. Such arguments routinely follow a queer course: cropped as mild mid range theory in the tradition of Talcott Parsons, hastily distilled into dry social laws in the manner of Malthus, and reintroduced into a Hegelian history as a teleological force to intoxicating effect. The ideological operation lies in that the presently limited phenomena casts its amplified shadow on the smokescreen of the future, so that its proponents can claim that history is on their side. Söderberg (2013) rightly points out that this is a case of technological determinism as a collective action frame, all the more interesting for its claims of apoliticism. The paradox is obviously that if the future is predetermined by technology, why people have to fight for it?

Shirky develops a useful scale of being organised, where he places *collaborative production* between *sharing* and *collective action*. The differences are that the former (sharing) does not necessitate collective decisions and producers largely retain credit over their own creations, while in the latter case (collective action) the cohesion of the group is critical to its success, and this shared responsibility ties the individual's identity to the group. While all these can be interpreted (and are interpreted by Shirky) on various layers from informal everyday practices through the formation of civil society to business and governance practices, it is suggestive to see the progression from sharing through collaborative production to collective action as a metaphoric move from the functions of civil society through capital to the state.

Politically, Shirky's argument is indicative of the trend in the peer production literature to comment on the collapse of the modern institutional grid, where broadly understood civil society not only challenges the capital and the state but increasingly takes over some of their functions. This tectonic, historical shift have been noted by various theoretical currents and understood in various terms, from contemporary interpretations of Lenin's dual power concept (Shantz 2010, 156–158; Dominick 2014) through Tronti's social factory (Cleaver 1992) to Foucault's understanding of biopower (Foucault 2010). It is notable how a similar analysis have been variously proposed as a political strategy for social subversion and a critique of a crystallising regime of exploitation and oppression. As critiques featuring in the last subsection warn, such transformation is very much a double edged sword. In revolutionary terms it can be seen as the advance of anarcho-communism amidst subsequent cycles of capitalist accumulation, the articulation of a germ form (as Stefan Meretz likes to put it) – yet at the same time it could be the creeping counterrevolution which provides the justification for the next regime of exploitation developed organically from neoliberal doctrine. This threat is a thread which runs through the current work.

Yochai Benkler can be considered the founder of the field, introducing peer production to a more narrow specialist but multidisciplinary academic audience (2006). As his colleague James Boyle notes (Boyle 2009), he presents powerful normative arguments for peer production as a viable, efficient, and non-coercive way of organising the economy. His definition of the phenomena is accepted and extended by a wide range of authors. In his interpretation **peer production refers to computer-mediated, massive online collaboration based on the principles of free association towards a common goal**. Its organisational form is characterised by a number of key factors, including *modularity*, *granularity* and the *easy integration of contributions.* Complex problems are broken down into simple ones, so that contributions can range across a wide scale according to the diverse motivation of the participants, brought together into a functional whole: modular architectures; granular participation and expertise; and unfinished artefacts. His observation about the political dynamics has also been largely accepted in the literature, including the key claim about democratisation that despite informal and formal hierarchies in peer production projects, there is no chain of command, monetary ties or coercion. His argument rests on the enabling nature of the Internet which is evident in his principle examples of the Linux kernel and Wikipedia (software and knowledge production). Notably, *he ties the possibility of emergence of peer production to immaterial production and the specific infrastructure of the Internet.* As the core tenets of peer production, these notions are used and examined along the thesis – through the crucial question of how far they play out in actual material practices.

Notably, Benkler's arguments are the offshot of a common political and research programme carried out by a generation of high-profile American lawyer-professors, including Boyle, Eben Moglen (adviser of the Free Software Foundation), Lawrence Lessig (co-creator of the Creative Commons licences) and Benkler himself. The trajectory of this project is important to understand the intellectual hinterland of peer production. Briefly, it was sparked by early conflicts between hackers and the state on the one hand; as well as hackers and capital on the other hand. The former have been epitomised as the *crypto wars* culminating in the declassification of strong cryptography as export-regulated ammunition, giving us the Pretty Good Privacy software (PGP, by Phil Zimmerman). The latter have been called the dawn of the *Golden Age of Hackers* by Levy (1984), involving the commercialisation of academic and hobbyist research with debates about Non-Disclosure Agreements and software patents, giving us the General Public Licence (invented by Richard Stallman). The legal strategy involved in both was the framing of programming as free speech protected by the Fifth Amendment. These concerns converged on the critique of copyright in general and the resistance to the extension of copyright terms in particular, which met with considerable organised resistance from rights holders.

The twin concepts of peer production as the process, and the Commons as the product constituted the middle phase of the project, when characteristic practices were synthesised and generalised into principles as a positive contribution to society with potentially struc-

tural implications. It can be argued that this direction of development to broaden the theoretical and political ambitions was inspired by early successes in general legitimation and particularly concrete results, as well as the defeats about copyright despite wide popularity which forced campaigners to develop a more systemic outlook. The project since moved on seeing the above named figures pursuing their own trajectories towards the realisation of more limited middle term goals. For instance while Lessig seeks to reform congress to address the problem of money in politics through a campaign fund (MayDay PAC), Moglen concentrates his efforts on enabling users to keep their personal data on small servers at home which would mean stronger legal protection. Whether for transparency or privacy, these efforts are justified by the ideas of peer production and the commons developed earlier.

Peer production therefore has a local context in the USA embedded in legal arguments about free speech, padded with techno-social conflicts and punctuated by landmark court cases (Coleman 2012). Notwithstanding such an embeddedness in a local context, the USA enjoys a global hegemony through which the voices of its internal critiques can also be heard, particularly on the Internet. Therefore in further discussions in many cases we deal with the "domestication" of these ideas as they are disseminated geographically through hegemonic infrastructures such as the Internet. In many of those cases the transition from software through culture to politics follows much swifter than in the organic development of the USA story. Given the roots of peer production discourse in conflicts between hackers and larger social structures, contemporary hackerspaces are arguably privileged sites for understanding how these practices developed, and how these theoretical ideas and political ambitions play out in practice.

Michel Bauwens dedicated his life to evangelise peer production and package peer production theory and praxis for a range of different audiences. For instance he is notable for promoting Peer Production for Marxist scholars in *Capital & Class*, for young entrepreneurs on his speaking tours, and most recently to governments through the FLOK Society project in Ecuador. His main contribution, however, is running the Foundation for Peer-to-Peer Alternatives and its knowledge base, the P2P wiki. My own work has greatly benefited from these efforts. Such a synthesising work of course presents its own difficulties — sometimes peer production appears like a solution looking for problems. It is also hard to do justice to such a diverse output since there is no one book or other definitive source. For this reason I mainly draw on Bauwens (2005) and Bauwens (2012), the first of which is a summary of his early understanding from a pseudo-Marxist perspective and the latter a summary of his current thinking oriented towards the capital and the state.

For Bauwens, **peer production is paired with peer governance and peer property, where the production is the input of governance and the output is property**. I find this hard to follow and rather stick to the idea of a single phenomena which the theory of peer production grasps as a *process* (production) and the theory of the commons as a *product* (property). Both have multiple aspects including legal and governance issues. Moreover, for Bauwens peer production and the other two concepts denote analytical and policy frameworks which have direct application to civil society, the state and capital. *This outlook adopted from developmental sociology is not productive for answering my research questions, but I largely follow the associated methodology of looking at peer production phenomena as it unfolds differently across the modern institutional grid, as well as Bauwens' central tenet that the significance of peer production lies in how it transforms that very grid.* The latter notably coincides with the central concerns of the two other seminal scholars, be it Shirky's amateurisation thesis or Benkler's advocacy of copyright abolition. On my part I focus more strongly on the social conflicts and divergences than the interest in the harmonisation of organisational, policy and accumulation regimes.

The common thread running through Bauwens' writing is the emphasis on peer production as a new way to allocate resources which is enabled by having an infrastructure in place which was not there in previous epochs. Such infrastructure is mainly technical, and secondarily legal and cultural. It supersedes capitalism whose hegemonic form of resource allocation is what the author calls Market Pricing. Peer production as a reality today is thus a prefigurative phenomena, "the kernel of the new in the shell of the old". Similarly to the classic conception of communism, peer production is thus clearly positioned on a modern historical timeline, a teleological arrow of time where technology inevitably develops, opening new possibilities for human progress. As discussed further on, this is the basic definition of technological determinism in Science and Technology Studies.

It is instructive to take a brief look at how Bauwens' understanding of peer production developed over time. **In the early texts peer production is an autonomous system for the production of immaterial goods that lacks its material basis.** Bauwens states that "at present, peer production offers no solution for the material survival of its participants" (2005). Therefore, "the key question is: can peer to peer be expanded beyond the immaterial sphere in which it was born?" (Ibid.) I identify two crucial points here: one is how far peer production can operate independently (autonomously) from other regimes, and the other is about the generality of peer production. I take up both questions in the course of the thesis, the first regarding hackerspaces as collaborative social formation and the second regarding open hardware as peer products.

The late Bauwens arrives at different conclusions. **Peer production cannot function as an autonomous system with its own agency, but has to be integrated into the modern institutional grid.** The title "Blueprint for Peer to Peer Society: The Partner State & Ethical Economy" makes this clear: the partner state is providing the infrastructure for peer production while the ethical economy is an entrepreneurial coalition which sustains producers economically while making products available as a commons – a corporate commons, with Doc Searls' expression. On the other hand, peer production moves from immaterial production to an organisational framework for quasi-general production. In 2012

Bauwens announces that a “new way to produce is emerging. By this I mean: a new way to produce anything and everything, whether it is software, food, or cities.” Remarkably, these conclusions did not develop in an anaerobic chamber of theoretical thought, but in conjunction with parallel practical efforts which sought to integrate peer production into the organs of the state and capital and simultaneously expand its scope beyond immaterial production.

My problem with the **post-autonomous interpretation** of peer production is that it does not take into account how peer production as a model is transformed by integration into modern institutions. Benkler is clear on this point when he makes a distinction between commons-based peer production (like Wikipedia) and hybrid implementations of peer production (like YouTube). In the subsequent chapter on hackerspaces I show how the early hacklabs which were conceived as autonomous from the institutional grid were superseded by the hackerspaces which accept and advocate a tighter integration, and attempt to analyse the differences between these strategies and historical periods. Furthermore, the case studies which follow offer an opportunity to look at the interaction between relatively autonomous peer producing projects and larger social structures like the capital and the state. In any case the post-autonomous interpretation is historically sound in the sense that there is a diagnosable shift during the last decade from isolated projects to deeper integration into society, the market and the policy space.

The problem with the **generality thesis** on peer production is that despite the enormous efforts of practitioners, most relevant cases at this point in time are little more than pilot projects or at best nascent niches on the market (like 3D printing), so conclusions on the generality of peer production may be premature. Again, we have to be attentive to the characteristics of peer production according to the product in question, because models may differ considerably. I argue that the study of open hardware – discussed separately in the next section of this chapter – is strategic for such inquiries because it incorporates both hardware and software aspects that can be compared. While the generality case have been more or less convincingly argued in the last years of peer production literature it is not yet a strongly established historical reality – but then again, “hardware is hard”, so it was not even expected to happen at the same speed that peer production in software development achieved industrial relevance.

All in all, while for Shirky peer production is a way to organise without organisations (a new way of group formation), for Benkler a production regime and for Bauwens primarily a resource allocation framework, all agree that it is most interesting as a medium for the extension of participation which reconfigures the mobilisation of expertise. Bauwens writes that “P2P does not refer to all behaviour or processes that takes place in distributed networks: P2P specifically designates those processes that aim to increase the most widespread participation by equipotential participants.” (Bauwens 2005) Moreover, all highlight the significance of such a paradigmatic change in organisational patterns for the theoretical reinterpretation and the practical rearticulation of civil society:

again, as Bauwens points out, civil society is so far called non-profit and non-governmental exactly because it is seen as a derivative sphere, whereas the ideas of the partner state and ethical economy show how those other two domains can be adopted or are adopting to the first.

3.2.1.3.1 Points of contention. Now that a discussion of the mainline of peer production theory is available to the reader, I summarise some points that a social constructivist (STS) point of view can bring to the discussions around peer production. I make three points and an additional note, going from the most general to the specific points of constructivism. My points are not very original – indeed they are merely more than the application of core constructivist principles. For the same reason I am not tying them to specific STS authors because any appropriation would be questionable and probably unfair.

Firstly, as noted briefly earlier, much of the arguments presented above depend on a more or less well hidden technological determinism. Technology is something with a life of its own, characterised by incessant development that brings gradual social progress. Shirky's argumentation is a case in point. The second subtitle on the cover of some of the editions states squarely that "revolution doesn't happen when society adopts new technology, it happens when society adopts new behaviours." While such gestures are reiterated throughout the book, there is no other explanation whatsoever why society adopts new behaviours, except for the adoption of new technology and its inherent properties. In other words, technology in not an endogenous variable depending on the state of the system but an exogenous variables which defines its environment – and even worse, this variable increases automatically and proportionally with time. According to the main tenets of STS, technology development and adoption is dependent on social factors, and answers to contemporary social conditions – or it never gets developed or adopted.

Instead of accepting the emergence of new technologies as a given fact, we have to look at the conditions of possibilities for them to be conceived, developed, and perhaps eventually adopted by society. Similarly, sheer efficiency – e.g. what Shirky's account emphasises: that it works well – is not a good explanation, because efficiency itself is a concept defined through struggles between the people themselves and the materials involved. Instead of "emergent" and "new" technologies and phenomena, we have to be able to conceptualise these issues in terms of "socially constructed" and "path dependent" technologies and practices. This is what is meant by social constructivism in contrast to technological determinism. Therefore, we have to contextualise the development and adoption of both new technologies, and most importantly the theories and practices of peer production itself, in social history.

Secondly, if we accept that neither technologies nor societies develop on their own, but through social, economic and political struggles, we have to recognise that it is particular social groups who develop technologies and the same for those who adopt them. The term *relevant social groups* in the Social Construction Of

Technology (SCOT) stream of STS seeks to capture this. These groups often have differing interests in general and different ideas of what the technology should be in particular. Even the social groups who do not participate in the development and refuse to adapt a particular technology will be affected by proxy, and affect the technology through their absenteeism. Technology research and development, as well as adoption and appropriation, is a constant struggle for the functionality and meaning of the tools at hand. Especially research on any "architectures of participation" as peer production is, should include, at the minimum, an account of who the participants are, who is not participating for what reasons and which are the division lines between the participants. Such an approach on one side explores the power dynamics behind the technologies and the social dynamics which form around them, and at the same time presents the contingency of their trajectory in terms of social meaning and technical functionality.

In contrast, all authors above refer to users as an undifferentiated mass, – in Shirky's words, a "society" – which reproduces the universalist fantasies of classical liberal thought. While it is unquestionably true that peer production methods lower the barrier of entry for participation and the level of necessary expertise to live with that opportunity, these very changes benefit certain social groups more than others, and interpreted by each particular social group based on their previous practices of meaning-making. It is easy for theorists to disregard such nuances because the discourse of participants most often ignores these differences too: a commons is nominally open to the general public and there are no rules barring anybody from accessing it. The emphasis on quality in collaborative development often comes with the assumption that the right ideas, implementations and designs will automatically win at a receding future point in time, as a result of a kind of technical Darwinism. In this sense it is interesting that given the large number of works in peer production theory which operate with the concept of revolution, the notion of social conflict is almost entirely missing. An example is the social evolutionist thought of Bauwens which concedes that a reactionary part of the bourgeoisie would challenge peer production in case it would become a historical force, yet trusts the superior architecture of the peer production model to outperform its competitors whether in civil society, business or state organisation – because history is on the peer production side.

Thirdly, peer production appears as a generic model which can be identified in the most varied settings, independent of space and time. For a more rich characterisation of peer production efforts, a geographically and historically varied account would be necessary. What do we know about the various stages, impasses and blind allies through which peer production developed, or its reception and appropriation in different geographical settings? How does the peer production of the 1990s differ from the peer production in the second decade of the twenty first century? Is there a peer production with Chinese characteristics? How does the fact that peer production theory and practice has been most strongly developed in the United States affect peer production and its appropriation elsewhere? Reading the above works sometimes gives the feeling that the action takes place in a geo-historical vacuum chamber where the laws of social physics are observed, or that Shirky's evocative New York setting stands in for the rest of the world. Fortunately in the specific literature of hackerspaces there is sometimes a keen awareness of social geography and social history, thus the adoption of the hackerspaces model in various regions of the United States and its uptake in East Asia or Latin-America has been explored enough to make a study of North European hackerspaces possible, while the geo-historical origins of both free software and open hardware have also been charted — the task is only to situate peer production in time and space.

To summarise, a Science and Technology Studies look at peer production have to treat its subject as *historically constructed* in geographically diverse settings through the specific participation and expertise of relevant social groups.

3.2.1.4 The economics of peer production: Transaction costs theory

Most works treating peer production include a chapter dedicated to transaction cost theory, which lends peer production its economic rationale. Here it is read without attempting to verify the relation of these theories to the actual practices on the ground (for lack of data), and without verifying transaction cost theory as a valid approach to economics (since it would be out of scope). What we are interested in here is rather the function that transaction cost theory plays in the peer production literature. In a way transaction cost theory is Benkler's answer to the three original puzzles of peer production: (1.) why commons based peer production works on a significant scale; (2.) why increasing number of people participate for free; (3.) does it mark a change in human nature? Before presenting transactional costs theory and the way it is mobilised to account for peer production phenomena it is useful to take three notes which bring it in the context of problematics engaged in the present study.

First, peer production phenomena is generally puzzling, but it is particularly puzzling for lawyers (like Benkler), economists (like Paul David) and businessmen (like Tapscott & Williams, see later). That is because for them it is hard to imagine people organising without laws, monetary transactions and profits. Difficult to theorise market agents that do not follow their individual, economic, rational self-interest. Complicated to understand how production can be socially useful before it is economically useful and legally binding. Nonetheless, these are generic (e.g. ahistorical) properties of any class formation (bourgeoisie or otherwise), class struggle, or even simply social movements. People often do organise themselves informally. Sacrifice themselves for ideals. Produce new social relations for the hell of it. If anything is new about peer production, it is the historical conditions which allow such things to develop into a whole cluster of significant social conflicts. Roughly put, these conflicts challenge the capitalist mode of production in the realm of law (copyright and surveillance), the market (the commons) and business (cooperatives).

Second, the above questions do go against the ground rules of both liberal economics rooted in the methodological individualism of self-optimising rational agents, and even some tenets of political economy built on the notion that workers are dispossessed of the means of production. Therefore transaction cost theory is an attempt to bring the peer production phenomena back to the realm of economic rationality and to a certain extent, political economy. Such attempt is successful if it can account for peer production in rational terms micro-economically and macro-economically. As we will see Coase's contribution can link microeconomics and macroeconomics and it can be used as the basis to argue for a rational account.

Third, following up on the critique of technological determinism in the previous section, these questions form a slippery slope to fall more deeply into technological determinist arguments. Some theorists cannot construct a historical account of the structural transformations in capitalism, but so they are left with the notion that what divides the present from the past is an inevitable and irreversible jump in technological progress. If all we can say is that "new technology" makes production more efficient, driving down transaction costs, which in turn changes the motivations of people and leads to a morally superior human race, we succumb to the worst narratives of modernity (as Rifkin does, at the end of this section).

Now that the stakes of the transaction costs argument are clear it is possible to proceed to its presentation. Ronald H. Coase is the Marx of transaction costs theory and his 1937 paper *The Nature of the Firm* (1937) is its *Capital.* The chain of references in the peer production literature point to Benkler's long paper *Coase's Penguin* (2002) which lay the groundwork to his magnum opus discussed above, *The Wealth of Networks* (2006). Transaction cost theory was born from the intercourse of economics and law, developed from Coase's work by Oliver E. Williamson (1983; 1998) to the field of new institutional economics.

Coase's original question is how big an ideal firm grows, which quickly leads to the question of why firms (cartels, monopolies, etc.) are necessary at all if according to liberal economics the market is the ideal resource allocation mechanism? The answer is that under a historically specific legal and technical regime (later called the institutional environment, Williamson 1995, 211), some operations are more economic to undertake in the protected realm of the firm then in the open market.

The argument itself about firm size is made in three steps. First, it is posited that all operations have marginal costs, which is defined as the overhead payed in addition to the actual price of the goods, services, right, etc. acquired. DeLanda compared this (2001) to friction in physics. Second, *marginal costs* are different on the market and in the firm: on the market where the price mechanism is the medium or mediator, they are *transaction costs* like the very discovery of the appropriate market price, legal costs of establishing and enforcing contracts, etc. while in the firm where the medium is hierarchical command and control, they are *organisational costs* like management and coordination mechanisms, work discipline, etc. Third, organisational costs grow together with the firm while the market scales infinitely. This in itself explains why we see big corporations as slow behemoths while big markets appear as bustling. Following DeLanda's clue, we will continue to see an imagery of acceleration unfolding in the mechanics of transactional costs theory.

The final conclusion – also called Coase's Law – is that a firm ideally grows until the organisational costs for its operations outgrow the transaction costs, at which point it is more economic to outsource them. The moral of the story is that the market is only one allocation mechanism which works in conjunction with others – like the firm in this case. As DeLanda notes, legal theory meets economics in the domain of organisational sociology here since the logic of the explanation rests on the idea that it is not only goods which change hands but the various rights to use and direct a variety of resources, and such movement does not occur without costs.

Where peer production theorists – still largely following the lead of Benkler – come into the discussion is the historical specificity of Coase's argument. They argue that in late capitalism transaction costs fall rapidly because of technological progress which transforms the institutional environment where firms operate. Such a fall is so dramatic that it tends to zero, where certain operations are not even worth to take into the market, let alone the firm. The result is the emergence of peer production as a submarket domain where information and knowledge are transferred with minimal friction, therefore the mobility of knowledge capital approaches the infinite. As Clay Shirky explains, "Large decreases in transaction costs create activities that can't be taken on by businesses, or indeed by any institution, because no matter how cheap it becomes to perform a particular activity, there isn't enough payoff to support the cost incurred by being an institution in the first place." (2008, 46) However, the social conflict arises when it turns out that the acceleration which is driven by technology is not followed by the other component of the institutional environment: the legal framework. This is the field where the aforementioned triad of US lawyers lead an army of free software, free culture, free society activists and practitioners against copyright as the pinnacle of monstrously outdated property relations.

In the same spirit Benkler states that **"Transaction costs associated with property and contract limit the access of people to each other, to resources, and to projects when production is organized on a market or firm model, but not when it is organized on a peer production model."** (2002) It is thus clear that (1.) peer production is distinct from the market and the firm, (2.) property (the basis of firms) and contract (the basis of the market) stand as a limit to the allocation of resources, (3.) and therefore the current regime is detrimental to both social and economic development. The normative program drawn from this syllogism then is to transform firms (e.g. capital) into what Castells calls the *network enterprise* (1996 Chapter 3), the legal frameworks (e.g. the state) into what Bauwens calls the *partner state* (2012), and reorganise economic life according to what Boltanski and Chiapello identifies as the *project order* (Boltanski and Chiapello 2005). Analysing hackers as a social movement, Söderberg (2013) rightly points out an internal contradiction between the overall narrative of a teleological history driven by technological progress (in conjunction with economic rationality) and the actual campaign of social mobilisation to actively thrive for the victory of such a necessary truth. Paradoxically, in the ideological project of peer production, technological determinism plays the role of a framework for collective action.

While the movement cannot overthrow the current property regime at once, what is already possible within the confines of the present system is to reverse it in a subversive way. As Steven Weber notes before Benkler, copyleft licences like the General Public Licence (GPL) use copyright as a distribution mechanism rather than as a restriction mechanism. This marks out a distinct territory for the growth of peer production, animating the other element of the institutional environment (i.e. the legal framework in addition to the technological component). While Weber recognises decreasing transaction costs as a structural transformation, he draws attention to the legalistic foundations of Coase's economic theory: "My argument here simply is that shifting property rights can and will likely destabilise the foundations of existing cooperative arrangements and institutions, and possibly in more radical ways than do changing transaction costs." (2004, 257) In essence then Weber complements changing property relations as an independent historical process which should be considered along the decreasing transaction costs. **Crucially, this idea is picked up by Benkler two years later in his own magnum opus (2006), but in the formulation of the latter scholar, this second line of attack about changing property rights is an opportunity or even necessity which is opened up or determined by the falling transaction costs.**

Both authors point out that the real drivers of peer production are the type of transactions which have never been viable on the open market, but thanks to the decrease in transaction costs they become practicable now as peer production operations. That is, contributions so small or so informal that they were never worth or worthy to be compensated economically, become practicable and significant. In this moment a wide and wild variety of motivations (both intrinsic and external) come into play, which allow the simple availability of information about tasks to take the place of the price mechanism or the hierarchical command-and-control structure. The sphere of significant economic activity outside the firm and the the market which opens up at this point is the sphere of peer production. Weber seconds Benkler in arguing that this point is key to the success of open source projects, since they can tap into a larger part of human creativity then proprietary models. The associated innovations in property regimes keep up such motivations and therefore stabilise participation because they decrease the risk of any one rouge actor drawing an enclosure around the common product, taking exclusive benefit from it. A risk that also both authors see clearly is that if tasks are self-selected according to the self-perception of the participants that they have the necessary expertise to complete them, it is necessary to complement such freedom with increased measures for quality control – which ironically is an eminent marginal cost of production. Benkler's counterargument is that quality assurance can itself be organised collaboratively, and Weber refers to *Linus's Law* documented by Eric S. Raymond that "given enough eyeballs, all bugs are shallow." Indeed, in parallel to its growth to the largest collaboratively organised software integration project (comprised of more than 37500 software packages), Debian GNU/Linux have seen an disproportional increase of activity around the Debian Quality Assurance Team. This has been important also because as once again both authors point out, the integration of contributions requires unusually skilled participants. As I argue later on, at some point expertise becomes the limit of participation, which means that the unwashed masses are closed out, while at the same time those few with the the right skill set have more and more systems integration work to be done. This could become a crisis where low transaction costs do not do not necessarily help, since integrators are the centre of the network can be easily flooded with contributions from all the corners of the Internet.

Where Benkler and Weber really diverge is that the latter displays a sensitivity to political economy, which allows him to identify the ideological drive of the transactional cost argument (2004, 255), alluded to in the introduction to this section:

> The ability to move information around the world without friction has been deeply associated with a market metaphor, even more deeply a market-based ontology as a way of seeing the world. Two related things point in this direction: decisions being pushed down either to the individual or to the machine on a case-by-case basis, and the massive reduction of transaction costs enabling those individuals (or machines) to find each other and agree to an exchange. And so the Internet has often been portrayed as a "perfecter" of markets, bringing a vision of efficiency ordered through "perfect" information (as economists say) and Coasian equilibrium arrived at in relationships outside of authority (Weber 2004, 255).

Here peer production is not a submarket phenomena but a veritable *supermarket*, reversing the Coasian formula itself. Since what Coase – a self-declared socialist (Bylund 2014) – found is that the spontaneous freedom of the market had to be complemented by the planning authority of the firm, simply because a completely open market would underperform. In his theoretical framework, firms are not ideal market actors – as adherents of Austrian economics would like to think – but exactly the opposite: their internal logic is the necessary complement to the logic of the market. Thus they have to "suppress the market mechanism" (quoting Coase) to correct "market failures" (with the words of Williamson, both from Klein 2013). Such a conclusion obviously sets a theoretical limit to (neo)classical economics. How Coase intended his arguments is unclear, but it is interesting to consider that that he listened (disapprovingly) to Hayek's lectures at LSE, whose *Economics and Knowledge* (1938) appeared the same year as Coase's own *The Nature of the Firm*.

Reading Weber's paragraph in the context of both Coase's own time and the structural transformations in late capitalism usually treated under the rubric of neoliberalism (Harvey 2005), it is easy to see how peer production can be part of an overarching ideological project. Indeed, peer production's mobilisation of sub-economic incentives opens the way to the radicalisation of *classic* neoliberalism which absorbed the state (and with it the property regimes) into the market logic. Identified as the Californian ideology (Barbrook and Cameron 1996), the supermarket extends the mandates of the market ontology through cybernetics far beyond the confines of economics and politics to a general ordering principle of everyday life. Extensively, it ensures the frictionless mobility of capital (through the free reign for the primitive accumulation of openly licenced goods), while intensively, the deepest exploitation of human creativity (through the real subsumption of fully mobilised human productive capacities).

As we have seen earlier, Shirky's results also show how peer production sociality increases productivity beyond the limits of the market by tapping human instincts – an argument supported by transaction costs theory: "[r]idiculously easy group-forming matters because the desire to be part of a group that shares, cooperates, or acts in concert is a basic human instinct that has always been constrained by transaction costs." Marx already noted that productive cooperation as a positive anthropological factor is mobilised as the wage relation turns work into labour, where surplus labour is expropriated by the capitalist, but peer production presents a model in which work discipline and monetary incentives are replaced by ideological effects and technological interfaces. Open licences do not necessitate any more that workers have to be alienated from the labour process or disenfranchised from the means of production. On the contrary, as peer production advocates Adam Arvidsson and Nicolai Peitersen argue, (Arvidsson and Peitersen 2013) their productivity lies in their ability to establish meaningful ethical relations to each other. Weber aptly concludes that "In this kind of discourse… the perfection of markets and the realisation of potential communities are theoretically identical."

In effect, then, **the significance of Coase's transaction cost theory is precisely that it identified the limits of the market**. *In contrast, peer production advocates use technological determinism to reverse Coase's argument and present it as the ideological basis for a market without limits. The attack vector which peer production advocates successfully exploit is that Coase rightly refers to the institutional environment which determines transaction costs as a historically and geographically specific formation. The technological determinist argument of the peer production advocates then is that the institutional environment changed simply because technology evolved.*

To come back to the three points in the beginning of this section, even though such an argument managed to bring back peer production phenomena into the realm of individual actors seeking their rational self-interest, it also restored faith in an essentialist reading of human nature, and at the same time it did not challenge technological determinism effectively. Managerial literature such as the pioneering and paradigmatic *Wikinomics* conveniently summarises such a line of interpretation:

> Strictly speaking, the law remains as valid as ever. But the Internet has caused transaction costs to plunge so steeply that it has become much more useful to read Coase's law, in effect, backward: Nowadays firms should shrink until the cost of performing a transaction internally no longer exceeds the cost of performing it externally (Tapscott and Williams 2006, 56).

On a very different level but with comparable naivety, aiming for long-term well-being rather than short-term profit-making, Jeremy Rifkin takes the same arguments to their logical conclusion. In his vision of the *zero marginal cost society*, (2014) technological progress inevitably drives down marginal costs to approach zero. This does not only lead to the end of work as in his previous works of "science fiction", but the end of capitalism itself – all without significant social conflicts. Rehearsing the reasoning of the first Do It Yourself 3D printer manufacturers, the makers of the RepRap 3D printer, he argues that self-replicating personal manufacturing machines will allow everybody to produce anything practically for free. Free home-manufactured goods will drive companies to bankruptcy since there is no demand for market products any more, and capitalist competition already made sure that profit margins are slim. Sharing then becomes a rational choice when goods are in abundance. In the final scene, autonomous robots perform the bulk of work – as in Karel Čapek's 1920 play *R.U.R.* (*Rossum's Universal Robots*) –, and humanity turns into an altruistic species thriving on knowledge and pleasure, living in harmonious collaboration.

In order to understand where such techno-utopias are coming from, we have to refer to Söderberg's reading of the RepRap 3D printer pioneers' discourse. (2014a) He manages to **trace back the ideas and ideologies behind the self-replicating machines to the Saint-Simonians**. Saint-Simonians are the bearers of *engineering ideologies*, crafted by a particular social group whose class formation took place during the French revolutions and the political-economical regimes which followed. In the initial period of capitalism this group was characterised by a conflated and confusing mix of proletarian and bourgeois forces. These "industrialists" – which in their leaders' original terminology included all the agents of the new order from factory owners through craftsmen to wage workers – were initially open to outright confrontation with the aristocracy and land owners, in order to overthrow the feudal regime by force. However, their military failures forced them to reconsider their position. By and by they broke with the nascent proletariat and engineers became central to shape their ideology. The socialism advocated by these latter-day Saint-Simonians therefore stressed the transformation of society "through the manipulation of the laws of nature, including the nature of fellow human beings", (2014a) through changing the technological conditions which they thought determined social relations. Building railroads for instance was interpreted and advocated by them as a way of transforming society for the better, without any class conflict and bloodshed. In Söderberg's reading they stand as the prototype for the ambiguous political stance of engineers, together with their similarly two-faced allegiance to disruptive capital. Such genealogy accounts for how peer production theorists can evangelise the "Second Industrial Revolution" driven by the development of technology, without mentioning social conflict which before was closely bound up with the idea of any kind of revolution. Therefore, the myth of modern progress which powers technological determinism thus took root in the ranks of engineers, reformers and entrepreneurs – novel social groups who began to build what today are considered public goods: roads, bridges and wells.

In transaction cost theory it is not clear if marginal costs like transaction costs and organisational costs occur within the firm itself – but at least it is assumed that the parties involved in the economic operations under consideration bear these costs, in one way or another. This is in contrast to externalities: costs of a transaction not payed by the parties involved in voluntary exchange, the classic example being a polluting factory which does not pay proper environmental taxes. In the case of firms monetising on peer production, however, we can even speak about positive externalities. This is how, for instance, IBM uses the Linux kernel in its products even though most of it is developed by third parties who does not have any formal economic relation to the firm. As with transaction costs, the full *ecology* of such a universe of exchange can only be considered by what I call *post-autonomous* economy, that is, economic inquiry which goes beyond the formally defined boundaries of its discipline. As the examples alluded to above demonstrate, such an economics is inevitably bound up with sociology and ecology, so that its proper subject matter is extended indefinitely, in a fuzzy way. Not only that, but these externalities and transaction costs that have hitherto not been taken account in socio-economic analysis which shape the structure of the system by serving as the limits of spontaneous market transactions and hierarchical firm organisations. This explains the novelty of these practical and theoretical developments, where marginal costs take centre stage. In a sense the study of externalities prepared the groundwork for a post-autonomous economics whose main insights come from adjacent fields like legal scholarship (the study of intellectual property rights) and organisational sociology (of peer production phenomena). While there is ample literature on externalities, it does not belong strictly to the topic, so I only mention two sources from the earliest and latest pronouncements. One of the latest is the *cognitive capitalism* thesis of Yann Moulier Boutang (2011), and one of the earliest is Bataille's general economy. (1991)

Even though similarly speculative, *Boutang's reconstruction of the past neatly complements Rifkin's dream of the future.* **His central claim is that in the third type of capitalism which developed since the 1970s the main source of value comes from positive externalities, which have not been taken into account properly by classical economists.** These externalities are mainly derived from knowledge, but any non-rival public goods like roads or free software are counted in. They add value to the production process, therefore participating in the creation of wealth. The share of externalities in production increases exponentially from the 1970s, propelled by a similar decrease in transaction costs which allow them to be created and circulated. However, for Boutang unlike for Rifkin, these structural changes all lead to an overall social decline and the deepening of exploitation, as more and more of social relations are valorised and captured by capital, colonising the life world and leaving next to nothing of authentic human life. His theory avoids methodological individualism but nonetheless falls for humanistic essentialism and social determinism.

If Boutang reads Rifkin backwards, then we can safely say that Bataille reads the whole discipline of economics backwards. The latter's *general economy* starts with the sun – a practically infinite source of energy – energy which is given away freely and in abundance. It is in this context that the activity of various human civilisations is considered. The problem is that however production is organised, there is always an excess of energy to be spent – and the technical term the author uses for spending it is *luxury*, which includes the potlach, war and other excessive behaviours. In fact the type of luxury characterises the society under consideration and its political economic situation. The unproductive, surplus energy which has to go to waste through luxury is called the *accursed share* (the title of the three volume study): "it is not necessity but its contrary, 'luxury', that presents living matter and mankind with their fundamental problems". The author argues that capitalism burns this energy in crises, crimes and wars – and it can be further argued that peer production phenomena serves as an alternative pressure valve. As we will see later in more ethnographic detail, there are good reasons to think that the capitalist system of production never manages to capture all of the life energy in our civilisation, whereas peer production can mobilise some of it, even if in . The advantage of Bataille's general economy framework over transaction cost theory is that it can explain the seemingly infinite productivity, as well as the seemingly infinite wastage in peer production, without speculating about the individual motivations of the participants (i.e. methodological individualism), or trying to bring a hidden optimality their efforts (i.e. rational choice theory), or ascribing the inexplicable to the advancement in the machinery (i.e. technological determinism). Finally, it is important to note how the treatment of general economy falls into a tradition of economic thought labelled *post-scarcity economy* – which however will not be further explored here.

As we can see, *transaction cost theory is mobilised in similar ways but sometimes for very different reasons by various authors in the discursive field of peer production.* These notions are especially useful for understanding how peer production is instrumentalised in the context of concrete profit-oriented or ideologically-aligned projects as a business model or the basis for political organising. Transaction cost theory lends economic rationality to peer production practices from the point of view of methodological individualism, and legitimises peer production practices as the right direction for unleasing the full power of the market. While that could work theoreticaly, in his foundational work Benkler distinguished commons-baed peer production and hybrid implementations which characterises the appropriation of peer production by capital. **Thus the economic theory that justifies the ideology and the business practices advocated in the management literature may look contradictory: according to transaction cost theory, pure commons-based peer production makes perfect economic sense, but business gurus like Tapscott and Williams (2006) explain that when it is actually put into use by capitalists, it has to be combined with capitalist modes of profit extraction and implemented as a hybrid model.** It seems that in theory peer production is a perfect capitalism, but in practice it has to be diluted in a tincture of capitalism as if it was something else. These discontinuities reflect the ideological function that peer production theory performs in respect to business practices.

3.2.2 Open source hardware

Open Source Hardware (OSHW) is the specific area of peer production practices that I investigate through looking at the scene of hackerspaces in general and two chosen case studies of small scale electronic hardware in particular. The discussion of OSHW is undertaken in two main steps here. On the one hand, I explore practitioners' efforts to define, and indeed, to realise the concept of OSHW through practical and philosophical efforts, as well as organisational and legal instruments. On the other hand, I summarise some scholarly contributions in the field of Science and Technology Studies which grapple with the concept, in order to point out the central theme of the debate: the question about the generality of peer production. **The generality question is how peer production practices can be applied to physical goods**, which as we shall see leads to many subquestions.

Similarly to the discussion of peer production above, rather than aiming for a complete overview of the state of the field I am instead focusing on the parts which are relevant for the questions I am asking in the dissertation. These translate to three limitations. First, just like in FLOSS, the history of OSHW can be told as the history of OSHW companies. However, since I am looking at practices of OSHW in hackerspaces, the entrepreneurial side of the phenomena is only important as far as it concerns the community. Second, OSHW have been shaped by prominent projects which realised some hypothetical tenets that the movement championed – or failed in novel ways that the movement learnt from. However, the goal of the present investigation is to complement the studies of high visibility projects with cases which are more descriptive of everyday practices. Third, I am interested in how everyday practices of making small scale electronic artefacts can contribute to a more rounded conceptual understanding of OSHW. Therefore, the accent of my overview falls on the conceptual developments around OSHW.

State of the art

It would have been an order of magnitude harder to write about OSHW a decade or so ago because the landscape have been much more complicated and it was difficult to identify significant developments amongst a plethora of recent initiatives. Now that more consensus seem to have emerged, Open Source Hardware Association (2012) provides a synthetic overview of open hardware concepts, organisations and licences in a historical setting. I do not feel that a detailed account is necessary to complement it, and my recapitulation largely follows the flow of that description.

Entering the analysis, I would like to establish a degree of periodisation that introduces a break between the contemporary OSHW movement and its historical antecedents. In Chapter 6 I consider in detail the history of hacking and its origins in hardware hacking. As Antonić (2014) – a legendary hardware hacker who is active in both the old and new hardware movements – stated, "Before hackers could program, they first had to build a computer." In the more documented and therefore influential history of the US scene, the Homebrew Computer Club (1975-1986) united such efforts alongside major universities. Before the advent of the personal computer, *phreakers* broke into telephone networks to explore them and share their experiences with each other. The lively phreaker scene did not make much distinction between hardware and software issues – or physical security, for that matter – because they were so intertwined with each other.

Subsequently, the invention, widespread usage and varied use cases of computers pushed hardware hacking into the background. The exploration of computer networks offered unmatched challenges to hackers and social conflicts unfolded around programs and protocols rather than physical artefacts and infrastructures. While hardware hacking continued unabated, Perens (2007) notes that when he established the first open hardware initiative in 1999, the domain name expired before the second application for his certification system – indicating an all time low in activity in that area.

The epochal shift came about in the middle of the first decade of the new millennium. I investigate in more detail the internal and external, hard and soft causes of hackers turning back towards hardware in Chapter 6. Here I would like to highlight a more narrow issue pertaining to the topic of this section: why licencing hardware specifically became a possibility. Ackermann – the author of the first OSHW licence (TAPR OHL) – offers a technological determinist explanation:

> when products were built by hand using point-to-point wiring techniques, the intellectual property issues raised were straightforward; no one questioned whether a chassis full of wires was a derivative of the schematic diagram. By contrast, today's development process for electronic products, particularly related to printed circuit boards, opens the door to numerous intellectual property questions. (2009, 212)

I refer to Ackermann's article which does a great job to explain the complex technicalities of computer-aided hardware design. *From a social scientific point of view the crucial point is the claim that the appearance of the second wave of hardware hacking is a question of a resurgent convergence between hardware and software.* **While in the early days of computing software had to take into consideration the implementation details of hardware and the specificity of target architectures, today it is exactly because of higher level of abstractions that even hardware design is half a question of software.** Automated routing algorithms can calculate the optimal connection patterns between electronic components, software simulators can run tests to verify the properties of proposed hardware architectures, and much more of the functionality can be implemented in flexible microcode rather than solid hardware than before. In sum, **OSHW necessarily came after legal instruments became fashionable in the hacker world – after the triumph of FLOSS**.

One may say that the social conflict around licences was part of the process which I call the institutionalisation of the hacker scene in Chapter 6. Along with the repression brought about by hackers disregarding the limits set by the authorities and the vendors (as Coleman and Golub 2008 note), the positive movement of hackers around FLOSS forced them to codify their strategic goals and values as precisely as the law requires. As a result, the four freedoms defined in the GPL became rallying points of the hacker community even while they are subject to internal debates between the fans of different licences. The solidity of support for the vision inscribed in the four freedoms within the hacker scene is shown by the fact that when it came to codifying a legal framework for the peer production of hardware, none of the groups that arose criticised the goals of the GPL: the only question was how to realise the same values in the case of hardware.

Stallman (1999) already stated that "circuits cannot be copylefted because they cannot be copyrighted." The reason is that programmers could argue that their products are creative expressions and such interpretations gained the support of courts, so that software fell under the category of literary works. However, machinery has always been protected by patents. Indeed, programmers fought hard on both sides of the Atlantic against the introduction of software patents, winning in Europe and loosing in the US. From a political perspective this put open hardware initiatives in a tight corner. On the one hand, the legal instrument at hand was patents. On the other hand, hackers hated patents. Summarising the disadvantages of the patent, Ackermann (2009) states that trademarks are too expensive (around 5000 USD) and slow (around two years) to issue for peer production practitioners to adapt, and in any case the community has such an aversion to them that it is inviable to promote adoption. Therefore, the situation called for a similar level of ingenuity as the invention of free software.

While taking on the specific goals and implementation ideas of people behind the various attempts at bringing together the companies and communities working on an open approach to hardware would be an intricate tale of a labyrinthine movement – obviously full of dead ends – it is largely unnecessary. Open Source Hardware Association (2012) provides a good summary and differences are not as significant as they may seem. What is interesting is the wide range of social backgrounds and legal strategies which were explored in this formative period of OSHW.

The first attempt came from the direction of Bruce Perens, a legendary figure of FLOSS known as the author of The Open Source Definition and co-founder of The Open Source Initiative (OSI). Perens proposed a certification for hardware that resembled the OSI model. Just as OSI issues a definition of Open Source Software and an authoritative list of Open Source Licences, as well as certifying individual software as Open Source, a similar organisation could preside over Open Hardware, effectively issuing certifications to vendors of particular products without using a special hardware licence. The project was initiated in 1999 and revived in 2007 but failed to get traction.

Another attempt by Open Hardware and Design Alliance (OHANDA) was to introduce a trademark associated with a logo. The logo and an OHANDA-issued serial number would be printed on marketing materials as well as on the circuit boards themselves to connect online documentation licenced under a copyleft licence with the device itself and help users to recognise open hardware. Despite the good ideas the project folded some time after 2010 and only a handful of serial numbers were issued. A further input arrived from the scientific community: CERN (European Organisation for Nuclear Research or *Conseil Européen pour la Recherche Nucléaire*) issued its own Open Hardware Licence. Their cited goals were knowledge transfer from publicly funded academic research to other areas; improving the quality of designs through open peer review, and decreased duplication of efforts through sharing results. It is easy to see how these goals aligned with the position of CERN as one of the most complicated organisations that produces hardware in the world – and the organisation that produces one of the most complicated hardware in the world. Unlike other actors, CERN could both easily legitimise the licence in the eyes of the community and actually enforce it in face of non-compliance by its host of contractors. While issuing an open licence addresses the problem of working with a high number of companies which are potential competitors to each other on the market, it also addresses the need of a highly international effort which requires a unified legal landscape for the high number of inventions that are discovered.

Interestingly, probably the most popular open source hardware licence have been the TAPR OHL, which predates the CERN OHL as well. TAPR stands for Tucson Amateur Packet Radio Corporation, a civic association of ham radio operators – which is probably as far as we can get from CERN in terms of open participation and the democratisation of expertise. Understandably, TAPR members concerns were less about unifying a heterogeneous landscape of academic and commercial contributors but about defending their works from cooptation by the market. Unlike CERN, they could not hope to control any of the hardware through contractual means, business relationships and reputation. Ackermann's solution was to make up a licence that protects only the documentation itself, which he claims can still ensure the four freedoms in relation to the tangible device too. If going through the documentation is a necessary step in the manufacturing process then it is a strategic site of intervention because copyright – and therefore copyleft – applies to it. In sum, certifications, trademarks and licences for only documentation or for both hardware and documentation have been explored in order to bring the four freedoms to hardware. The truth of the matter is that none of these have been tried in court or at least the legal standing of these measures is not well established.

Fragmentation is probably one of the reasons why individual projects failed to gain traction beyond a certain point, but also why the Open Source Hardware Association managed to bring almost everybody except Perens on board with their ecumenical approach that involved interested parties right from the start. For later analysis it is analytically significant to mention that the first meetings of the OSHWA took place in the Eyebeam hackerspace in New York. Eyebeam is the same place where the RepRap project transitioned from a European academic research project with an open source licence to an American startup company, as detailed in Section 7.1. The OSHWA organised meetings of OSHW enthusiasts, popularised the OSHW definition and logo, as well as launching an OSHW certification programme as recently as the end of 2015 (Weinberg 2015). Since the activities of the OSHWA seem to be supported by the majority of stakeholders, I use their term – OSHW – instead of the Perens-endorsed "open hardware". Such usage also reflects the fact that more actual hardware projects are protected by TAPR OHL than CERN OHL, and the former protects documentation as the source of the hardware. *Despite these efforts to find a legal instrument that could be effective at defending the rights of hardware users specifically, even the official position of OSHWA is not clear on whether it is better to use an established FLOSS documentation licence to protects designs or one of TAPR/CERN OHL.*

Which brings us to the next point, the argument that in the waves of mobilisations around opening up technologies, OSHW is still in a state of flux. The obvious comparison to FLOSS is useful again, since as Kelty (2013) argues the latter barely exists today as a transformative social force – it has been integrated to business models to a degree that any disruption of capitalist accumulation regimes is negligible. OSHW, however, is still an ambiguous object. I show several examples in the case studies about the various possibilities explored in practice about the relationship between the community and the industry. It is plain to see that such a relationship is even more crucial with OSHW since it is dependent on the factory production of at least the microchips involved. Hess (2005) described such a relationship as a product-oriented movement where object conflicts are bound to emerge because the community have different goals than the industry. On the other hand, not all waves of open source technologies are created equal. Even if the history of FLOSS is deeply intertwined with the history of companies right from the start, the FLOSS developments arguably took companies by surprise, so that the relative autonomy of hackers could be better leveraged. That happened to a certain extent with OSHW too, but given the alignment of FLOSS and OSHW ideals, the latter evolved in an entrepreneurial environment with ready made business models since its inception. It is quite plausible that this is one of the reasons why hacklabs which worked in an explicitly political (anarchist/autonomist) milieu did not jump on the OSHW bandwagon, amongst the other reasons found in Sections 6.1.3 and 6.2. The third wave of open technologies that hackers decided to take on recently is biotechnology, where participants are set on developing prototypes for products with a business model in mind (Delfanti 2013). These developments are further explored in Section 7.1 later.

In summary, what is important here is the following three points. First, **open technologies come in waves and present new problems so that the relationship between the community, the industry and the regulatory framework have to be gradually clarified and a set of agreements and best practices have to be established between participants.** Second, **OSHW is clearly less mature than FLOSS and more mature than DIY biology, which means that both the industry and the community have some understanding of the possibilities involved and experience in exploring those, even though these have not yet stabilised.** Third, **the degree to which market integration pressure can structure the culture and practices of participants in a wave of open technologies increases in every subsequent wave in proportion to the attention it received (early on) from capital – a measure which can be taken in terms of Venture Capital funding available in the given technology area.** OSHW is already well supported by a range of manufacturers, resellers, SME design practices – a whole ecosystem has grown up – without finding any quantitatively important applications comparable to the significance of FLOSS in contemporary computing. All in all OSHW is a niche field with major experiments already on the way.

Taking a handful of (high profile) OSHW projects, the present confusion is clear. The Arduino rapid prototyping microcontroller board protects its design files with a Creative Commons Attribution Share-Alike licence. Tinkerforge, a system of stackable microcontroller boards is licenced under CERN OHL. RepRap, the aforementioned 3D printer project choose the GNU GPL to protect their designs. The Helix_T turbine which is the subject of Kostakis, Fountouklis, and Drechsler (2013) uses a Creative Commons Attribution Non-commercial Share-Alike licence. Finally, one has to acknowledge that according to a 2013 OSHWA survey – which was not designed to be an academic research – almost half of the 1007 responders released what they understand as OSHW at least partially without any indication of licence. These quantitative results are echoed by qualitative assessments by scholars like Christian Siefkes who follow the developments of the field:

> Most open hardware projects seem to care little about the specific

> issues of hardware licensing. Most projects aiming for copyleft just apply a standard license such as the GNU GPL or the Creative Commons BY-SA license, apparently either not knowing or not caring that it won't apply to building hardware (2009).

The last point is what I demonstrate in different ways in the discussion of empirical results from the case studies, taking the Open Source Hardware Definition championed by OSHWA as a starting point for the investigation. I can already point to the relative lack of controversies about licencing articulated in the scene, despite the variety of practices and confusing about the effectiveness of the legal implementations of the shared values of the community. In pitting practice against the theoretical formulations of the definition, I show how **social relations take precedence against legal instruments in the specific context of the hacker milieu**. Again, hackerspaces appear strategic for the study of the peer production phenomena since they are largely protected from competitive pressure from the market and regulation or repression from the state.

Scholarly perspectives

The owl of Minerva spreads its wings only with the falling of the dusk – wrote Hegel, remarking that scholarship inevitably lags behind the historical situation. And indeed, practitioners did not wait for scholars to debate and decide whether peer production practices could work for hardware or they are only viable for software. Moreover, it seems that the question remained interesting – or even rose in relevance – for academics after practitioners began producing OSHW in bulk.

Kostakis, Fountouklis, and Drechsler (2013) is indicative of a trend of *messianistic scholarship*, using a single case study to prove three points of historical scale, namely that (a.) **commons-based peer production can be extended from the sphere of information production to other areas such as hardware (776)**, (b.) **peer production practices express internal and inherent contradictions of capitalist yet "might transcend the dominant system"** (775 referencing to Kostakis 2013), and (c.) **both structural changes can take place *without social conflict*** (778 referencing Hess 2005, 516). There are three problems with such an argument. First, it may be true that commons-based peer production can be extended from the sphere of information production to other areas such as hardware, but there is a clear discrepancy between the empirical results the authors obtained – design and desktop manufacturing prototypes of a wind turbine – and the conclusions they draw – that urban public spaces can be filled with turbines that serve as personal wind farms. With the signature gesture of technological determinism that Söderberg once called the "flight into the future" (2014b) they assume that all the advances experiences in the case study process will be systematically rolled out to the masses, while all the stumbling blocks will be solved by engineers. These are obviously unrealistic assumptions hampered by many problems from the brittleness of current filaments for 3D printer, through the speed of printing to access to public space by citizens in contemporary cities. Most importantly, as von Hippel states,

> [i]n physical product fields, product development by users can evolve to the point of largely or totally supplanting product development – but not product manufacturing — by manufacturers (2005a, 14).

The case study in Chapter 09 shows a failed attempt at breaking the manufacturing barrier by the local community of a hackerspace. The alternative to mass production at home is the distributed manufacturing vision prophesied by Anderson (2014) or Gershenfeld (2005) for Californian entrepreneurs and the third world and Dafermos (2014) or Rigi (2013) for developing countries and revolutionaries. The case study in Chapter 10 shows an alternative to distributed manufacturing through the democratisation of knowledge together with infrastructure, rather than merely manufacturing capacities which have actually occured in practice.

Second, investigating the transcendence thesis of peer production is as important as investigating the generality thesis of peer production, but it has to include the development of social relations in the account along with a structural analysis of capitalism in general and the contradictions that OSHW expose in particular – referring to the development of dead labour as the sole explanatory principle is not enough or effective. Symptomatically, even though Kostakis, Fountouklis, and Drechsler (2013) acknowledge the problem of capitalism for the systemic development of peer production practices in the theoretical introduction of the paper, their actual analysis of the case study does not contain any reference to the historical period in which the particular OSHW project and the means of production that allegedly made it possible exists. Therefore it is fair to say that the study is not historically situated, not to mention its lack of geographic specificity. Similarly, even though the authors are quoting Ackermann to the effect that "electrons are cheap, but atoms are expensive" (2009, 210–211) there is no mention of the contributions and complications caused by porting peer production practices particular to FLOSS and other areas of knowledge production to OSHW. It is fair to assume that such a technological transition would pose specific problems, such as the ones pointed out in the previous section. Third, the idea that technological change, engineering expertise and the democratisation of design can lead to structural change in a peaceful way is as deeply flawed as it is deeply ingrained in the entrepreneurial culture which frames much of the discourse around peer production. Söderberg (2014a) identified some of the first instances of such an ahistorical argument historically, tying it to the struggles of the Saint Simonians during the French revolution. It is telling that the narrative of bringing revolutionary changes to society by building bridges and laying train tracks has been adopted after the strategic defeat of the workers weavers' uprising in Lyon in 1831. Referring to Musso (2010) he concludes that

> the state repression that followed prompted the remaining Saint-Simonians to change their rhetoric and style of thinking. The role of struggle in the social transformation that they propounded was played down. Social change would instead come about through the development of communication networks, chiefly railways and canals (Söderberg 2014a, 5).

As the author continues to argue, the ideas of the Saint-Simonians are cut from the same cloth as the techno-utopianism of Adrian Boweyr who launched the crown jewel of OSHW: the RepRap project. Thus, the position articulated in Kostakis, Fountouklis, and Drechsler (2013) merely goes beyond the attempt to find theoretical legitimation for the visions of practitioners and lend political legitimacy to movements rallying around technological alternatives. Once again, these are worthy goals to pursue. Academics should be in solidarity with the movements and practitioners they see as progressive social forces, but scholarly solidarity should entail constructive criticism where it is found necessary. Otherwise it becomes apologetic, leaving behind the tenets of science and technology studies to embrace technological determinism.

A corrective movement is visible in Troxler (2010) who looks at actual practices in relation to the prospects articulated by the previous category of writing, arriving to three conclusions that balance the preliminary analysis provided by the previous authors. Troxler's analysis is valuable for the research project advanced here especially because he is taking up similar issues from the specific point of view of shared machine workshops, even if he is working in the Fab Lab instead of the hackerspaces context (for differences between Fab Labs and hackerspaces, see Chapter 8). So the question is what the extension of FLOSS practices to OSHW means for open collaboration organised according to the peer production model. First, he points to the lack of concrete results regarding the generality thesis of peer production:

> Indeed, despite the many academic discussions that support such a view, it is naïve to believe that open source software practices could be copied to and applied in the open design realm without any alteration, ignoring the constraints and opportunities that the materiality of design entails (Troxler 2010, 74).

Even if the applicability of peer production theories to OSHW production have been proved conclusively, there is little understanding of what differences does such a shift in the materiality of artefacts makes.

Next, referring to Make Magazine's guides to OSHW projects such as Torrone (2009):

> Torrone and Fried have shown how a regular and sizeable market has grown around open source hardware. Those open source hard-

> ware businesses clearly operate under market conditions and their production is not radically decentralized (Troxler 2010, 74).

OSHW therefore managed to create a new niche market for SMEs but it is doubtful whether the prospects pronounced by the likes of Kostakis, Fountouklis, and Drechsler (2013) ever materialised. Troxler's point is that OSHW did not manage to create hybrid business models interpreted along the lines of Benkler (2006)'s idea of hybrid implementations of commons-based peer production practices that would ensure the sustainability of community-based shared machine shops. Indeed, the results presented mainly in Chapter 7 show that the sustainability of hackerspaces rests upon the membership fees of hackerspaces. Where our analyses diverge is that Fab Labs explicitly aim for sustainability based on revenue from outside sources, while the Hackerspace Design Patterns of Ohlig and Weiler (2007) that provide a blueprint for these organisations explicitly advice relying on membership fees to sustain hacker clubs. As one member of the hackerspace in Budapest, Hungary (H.A.C.K.) put it, "the difference between the two places is that they pay you to go there, whereas we pay to go here" (dnet, hacker, personal communication 2010-04-11).

Understandably – coming from the Fab Lab context – the expectations of Troxler is that OSHW development should both allow a wide spectrum of collaboration and the democratisation of expertise while at the same time yield revenue towards the sustainability of shared machine shops. The results obtained by Söderberg (2010), reviewed later, show a clear contradiction in these two ambitions. However, the study of hardware hacking in the hackerspaces context allows for the exploration of OSHW projects that are not oriented towards creating commodity electronics, which may serve as a privileged environment for understanding the internal dynamics of peer production practices as they apply to open collaboration around small scale electronic artefacts. Moreover, if such a financial model seems to restrict participation to a privileged group of users, see Chapter 7 for the corrective social dynamics at work in hackerspaces. Throughout the dissertation I will argue that the open door policy of hackerspaces in practice creates more opportunities for participation in hardware hacking than the external funding of Fab Labs.

Finally, Troxler makes a point that is important for the current investigation in reference to Balka, Raasch, and Herstatt (2009b) and Balka, Raasch, and Herstatt (2010), presentations of a comparative research of around a hundred open design projects which is written up concisely in Balka, Raasch, and Herstatt (2009a). **Their conclusion is that there is a significantly greater diversity in OSHW projects compared to FLOSS in terms of openness measured using a wide range of variables** (Troxler 2010, 75):

> In 2009, their database consisted of 106 entries, 76 of which were truly open development of physical products, or open design. Open

> design as defined on that site is characterised by revealing information on a new design free of charge, with the intention of collaborative development of a single design or a limited number of related designs for market exploitation. … "the degree of openness differs significantly between software and hardware components, in the sense that software is more transparent, accessible, and replicable than hardware" (Balka, Raasch, and Herstatt 2010).

These findings give important clues for research design. On the one hand, since not all OSHW are created equal one may want to focus on cases where the design is really shared and the intention of collaboration is there with prospects of a common project. On the other hand, in order to better understand the OSHW phenomenon it is crucial to pay attention to exceptions and divergences from our expectations based on FLOSS practices, because they may prove to be structural properties of hardware-oriented peer production practices. Putting the two together, I chose case studies that push the envelope of current conceptions about OSHW by leaning towards an especially rigorous implementation of peer production practices that are at the same time diverge from known FLOSS patterns. For these reasons I introduce the concept of *unfinished artefacts* instead of OSHW progressively through the dissertation as the development of the arguments allow, in order to capture the insights that go beyond the present emic and etic understanding of OSHW.

If Troxler orients his research towards a balance between the community and the industry, a synthetic overview of the industry perspective is provided by the persistent and prolific investigation of innovation researcher von Hippel, most prominently formulated in von Hippel (2005a) – right in the year when OSHW took off according to observers (see previous section).

While investigating the democratisation of innovation, the analytical perspective of von Hippel is tied to the interest of corporations looking for a business model in face of changing social conditions, and thus lacks attention to the political economy of peer production. That is why our analytical perspectives are complementary as to the emancipatory potential of grassroots research, development and manufacturing. von Hippel is not interested in exploring the anticapitalist impulses articulated through peer production practices like Kostakis, Fountouklis, and Drechsler (2013) or making use of its market potential to maintain communities of practice as Troxler would like to see: he is interested in maintaining profit margins despite the democratisation of expertise – the question that Troxler formulates as "whether and how traditional businesses will be able to adapt to a new reality of real prosumer choice" (Troxler 2010, 80). Covering a wide range of media from surf boards through software to hardware, von Hippel identifies three prevailing business models which are *not* specific to the peer production of small scale electronic artefacts:

First, *"Manufacturers may produce user-developed innovations for general commercial sale and/or offer a custom manufacturing service to specific users"* (15, 126). Such a business model seems especially suited for OSHW since large-scale distribution needs both investment and logistics, therefore going through the market and giving monetary incentives for mass manufacturing is often viewed as the only solution for making not only the design but also the device widely available. The other option concerns access to factories once the manufacturing potential of peer production practices has been exhausted, as in the case of the r0ket device discussed in Chapter 09. Indeed, in order to demonstrate the viability of such business models the author cites Thomke and von Hippel (2002) which discusses precisely the kind of business relation the hackers in my case study stuck up with a company in their region:

> custom integrated circuits offer an especially good example of custom manufacture of products designed by users. More than $15 billion worth of custom integrated circuits were produced in 2002, and the cumulative average growth rate of that market segment was 29 percent (von Hippel 2005a, 128).

Second, *"Manufacturers may sell kits of product-design tools and/or* product platforms *to ease users"* (15, 126). While such practices are marginal in FLOSS because because as Anderson (2009) argues the price of software tends towards zero, they are actually the backbone of the niche economy growing up around OSHW. On the one hand, even if the market for kits and *product platforms* is not significant macro-economically, micro-economically it may sustain the kind of creative communities that Troxler has in mind or – as we shall see to some extent – the hackerspaces. On the other hand, hackers are one of the most significant social group on the consumer side, as users of toolkits for custom OSHW designs. Putting the two arguments together these kinds of business models seem to be more strategic than others for hackers, whether it is as producers or as consumers that they participate in the associated niche markets. For better or worse, Chapter 09 contains several examples where such a relationship between the community and the industry have been not only beneficial but crucial for the realisation of the hackers' dreams of peer producing small scale electronic artefacts that can serve as the basis of electronically mediated mass collaboration.

Third, *"Manufacturers may sell products or services that are complementary to user-developed innovations"* (15, 126). This is the famous IBM/Red Hat business model of selling support for FLOSS. Significantly, given the maintenance and repair requirements of OSHW products and their potential for incremental development, such a business model could be potentially even more lucrative in hardware than software. However, in fact such a market segment does not exist – a sobering fact that is probably due to the little use that OSHW sees in industrial applications. Once again, we are reminded that OSHW is nowhere near the widespread adoption of FLOSS products in all spheres of life from hobbyists through academia to the business sector. Therefore, I conjecture that the absence of such business models is a sign of the immaturity of peer production practices organised around the creation of small scale electronic artefacts and the infrastructures that rely on them. Chapter 10 shows how the flexibility and modularity of OSHW can lead rather effortlessly to the emergence of large technological systems in the sense of Hughes (2012) which on the other hand require continuous maintenance and repair, as well as incremental development for incorporating the ever expanding needs of users.

In summary, while the full range of business practices von Hippel identifies is encountered in the proceeding pages, none of them proves to be integral to the peer production of small scale electronic artefacts in the confines of the hackerspaces – a discovery that in turn would be important for distinguishing OSHW from unfinished artefacts. I return to the discussion and evaluation of these results in the conclusion (Chapter 11). For now it is time to ask how Science and Technology Studies can contribute to the debate around software versus hardware in studies of peer production practices.

Powell (2012) asks a research question on a higher level of abstraction which is closer to core Science and Technology Studies concerns: **"How have specific cultures and associated legal codes become associated with democratisation of media production?"** (Ibid., 3) She is also looking for "points of connection and divergence between cultures of software and hardware hacking, as a means of identifying further instances of democratic participation through collaborative production." (Ibid., 10) While these questions are the right ones to ask, she seems to lack the empirical results to draw conclusions of any consequence, other than pronouncing that "the question remains" (Ibid., 18) about a "new set of opportunities for the democratisation of knowledge" (Ibid., 23).

The virtue of such approach – which I try to adopt in the subsequent investigation – is that stepping away from technological determinist or in other words commodity fetishist arguments, it asks about changes in participation and expertise rather getting bogged down with the question of access to technological artefacts themselves. However, scholarship on the topic have to break three barriers in order to answer these questions effectively. First, research have to be based on empirical material documenting practices on the ground, especially in the context of everyday life, rather than mapping discourse based on press releases in a journalistic fashion. Second, instead of identifying potentials for the future, the results of the research should produce insight into the political-economic content of the present practices. Third, instead of searching for further instances of the same, the question is how current practices fare in a historical perspective in relation to previous practices: what are the social conflicts they articulate?

Söderberg and Daoud (2012) set out to answer the latter types of questions, if only hypothetically. For them, adding OSHW to the core technical repertoire of hackers constitutes a decisive political possibility. Before, what they call (following Schiller 1997) *information exceptionalism* – the thesis that information can be separated from its context and has the magic properties of a non-rival good – delineated hacker politics. Hackers rallied for information freedoms and against intellectual property, but their critique did not reach the beating heart of capitalism: property itself. Later on, hackers sought to extend their activities to making, using and developing tangible electronic artefacts. The authors wonder if breaking such a barrier could bring hackers to confront property itself, and develop a practical critique. Furthermore, on a theoretical level whether or not OSHW seems to show similar properties to FLOSS can serve as a litmus test of the information exceptionalism thesis.

Notably, empirical results show an opposing movement. The historical shift described in Chapter 6 from hacklabs to hackerspaces involved turning the critique of private property into the critique of intellectual property, exactly around the time (2005) when OSHW practices entered the scene. Can they bring back a more far reaching critique of the existing conditions into the hacker scene? Again, all we can say now is that for the moment and for the most part OSHW piggybacks on the intellectual property protections offered by FLOSS licences by protecting documentation and not devices. If anything, the case studies in Chapter 6 show a growing disdain for property issues.

Fortunately, Söderberg did conduct an ethnographic study of an actual OSHW case for his dissertation (2011). The Ronja project in the Czech Republic involved the invention of a device to replace wireless antennas for point-to-point high-bandwidth connectivity, using flashing light as an alternative communication media (Söderberg 2010). Users managed to design, develop an deploy several instances and versions of the Ronja in a practical setting. They shared the design openly and choose off-the-shelf components so that anybody with minimal engineering expertise could reproduce the artefact. Object conflicts arose however over the commercialisation of the collective invention. On the one hand, marketing as a commodity looked like an effective way to distribute the device. On the other hand, the proposed business model required changes in the design like introducing mass-produced parts and proprietary solutions. **These empirical results are interpreted in the framework of innovation studies to suggest that the social background and political goals of participants are important for theorising their choices in terms of technical architectures and compositions properly.** Such a notion obviously disrupt the smooth cooperation that von Hippel above posits between the community and the industry. (Another incidental moral to take away from the Ronja study is that hardware projects may actually be easier to study than software in terms of the social conflicts around the technical architecture and composition, since there are not so many abstractions involved as in the case of software, which means social scientists may understand easier what is going on.)

In summary, three questions seem to inform the scholarly perspectives on OSHW. First, *the generality thesis of peer production* is still not worked out in detail and its implications to the democratisation of expertise are not clear, even though business perspectives have been explored extensively (especially in light of Kostakis, Fountouklis, and Drechsler 2013; Powell 2012). That is why it seems compelling to look at peer production practices of small scale electronic artefacts in a non-business setting. Which brings us to the second question: *the relationship between the community and the industry is often configured differently* in the case of OSHW than in the case of FLOSS, with hackers and hobbyists appearing variously as consumers or producers of commodities, for instance (especially in light of von Hippel 2005a). I contend that answering such a question adequately needs to take into account the relative autonomy of hackers worked out in Section 7.7 later on, as well as the way hackers could articulate their relative autonomy in different historical moments (explored in Chapter 6). Last, keeping with the trademark Science and Technology approach of *avoiding technological determinist arguments may be a trap if the research question is formulated exclusively in terms of technical differences between artefacts* (especially in light of Söderberg and Daoud 2012). All these questions evidently informed the research design of the present investigation.

Open source hardware in the hackerspaces

Seeking to demonstrate the significance of manufacturing capacity vis-a-vis innovating users, von Hippel refers to his own tinkering efforts, reporting a frustrating experience that turned out clunky and ugly results because of the lack of infrastructure:

> In my case it was especially frustrating to try to build anything sophisticated from mechanical parts. I did not have a machine shop in which I could make good parts from scratch, and it often was difficult to find or buy the components I needed. As a consequence, I had to try to assemble an… (von Hippel 2005a, 122).

Many commentators (Seravalli 2012; Kohtala and Bosqué 2014; Kostakis 2014; Dafermos 2014) note that this is exactly the problem which hackerspaces are well positioned to address. **The peer production imperative is that if the production of tangible goods requires a tangible commons of equipment and communities of practice sharing tacit knowledge, then hackers come together and establish such infrastructures.** Of course these could be accessible for the general public, a requirement that necessitates the opening of the club model for visitors. Hackers address that problem through OSHW projects that ensure open participation and access to expertise – this is the topic of the case study of door systems in Chapter 10. It should not come as a surprise that technology enthusiasts working on FLOSS and OSHW projects will adopt the same principles of organisation to running self-managed community workshops. At the end, unfinished architectures (hackerspaces) can support the peer production of unfinished artefacts (OSHW).

However, von Hippel's observation remains a valid point of contention, e.g. that "The *economies of scale* associated with manufacturing and distributing physical products give manufacturers an advantage over 'do-it-yourself' users in those activities" (14, emphasis mine). The counter-proposal of peer production advocates is to build on *economies of scale* along the lines of Teece (1980) and

Panzar and Willig (1981). Economies of scale thrive on variety and not volume and claimed to bring benefits in comparison to economies of scale regarding factors such as their environment impact and the responsiveness to local users' needs. Therefore some advocates of shared machine workshops for development proposed distributed manufacturing as an alternative to the cooperation of peer production communities and the industry (Gershenfeld 2005; Dafermos 2014; Anderson 2014). My own results in Chapter 09 show that attempts at realising such a vision are doomed by the low volume of output hackerspaces can manage or, as seen in Chapter 10, the high level of expertise required for distributing practices rather than manufacturing capacity. The conclusions about distributed manufacturing versus distributed practices of peer production are drawn in Section 10.4.

As an end note, the review of academic debates and practitioners' efforts would not be complete without mentioning the emerging field of open source biology which is putting most of the problems considered in this section in a new perspective. Notably, as the latest addition to the hackers' core technical repertoire that is supported by the infrastructures offered in hackerspaces, DIYbio builds on advances both in the FLOSS and OSHW areas. Thus most DIYbio projects are at the same time OSHW projects with the inevitable FLOSS components thrown in. Thematically, the strategic successes in health applications from water quality (Wijnen, Anzalone, and Pearce 2014) to prosthetics (Dickel, Ferdinand, and Petschow 2014) bring hackers into an environment with another set of possibilities, like collaboration between shared machine shops like hackerspaces, modern institutions like hospitals and concerned individuals like patients. Moreover, the changed environment means a closer engagement with the state through regulatory frameworks pertaining to the safety of laboratories (Deldanti 2014) as well as public funding for grassroots research and development. These tendencies are clearly articulated in the work of Joshua M. Pearce, notably in Pearce (2012) and Pearce (2014). Since peer produced biological materials, knowledge and equipment is out of scope of the current investigation, it is not considered in depth, only as much as it is necessary to the understanding of the OSHW ecosystem and the hackerspaces scene (in Section 7.1).

3.2.3 Participation and expertise

Democracy and technology is one of the most popular themes in Science and Technology Studies research, featured in the titles of many calls and conferences, books and papers. This direction of research seeks to exploit the potentials made available by the problematisation of hitherto unquestionable scientific facts and the politicisation of hitherto neutral technologies by early social constructivist analyses. The broad assumption is that if science and technology are not exogenous variables any more, but subject to political, social and cultural bias, then scientific and technological production becomes a public matter of exercising power which should be subject to democratic control.

Expertise and participation are the concrete issues which are tackled in STS when addressing the overall theme of democratisation. Perhaps a possible definition of science and technology as social fields is that within these areas of human activity expertise is an important limit to participation. One of the recurring points later on will be how the same is true but in different ways of the hackerspaces milieu. In particular I am trying to show how hackers meet similar dilemmas than scientists and technologists, but negotiate the tensions between participation and expertise differently then their counterparts working in more institutionalised settings.

Interestingly, hacker culture grew up and developed roughly in the same period and with the same speed as Science and Technology Studies, and the explicit problematisation of expertise and participation is also a strong stream in hacker culture. Do It Yourself enthusiasts often produce designs of things which are not available to the average consumer (like drones); Free Software developers write programs which rival expensive “enterprise solutions”; and independent security researchers make public disclosures of bugs that were only known to powerful criminals and spy agencies. These diverse activities have in common the perceived effect of levelling the playing field for actors of various scales across the social field and therefore arguably contribute to the democratic project of modernity. Without analysing them further, here it is only topical to mention them to signal that STS shares common concerns with various tendencies in hacker culture, and therefore questions about expertise and participation are worth asking regards to the hackerspaces.

But what kind of questions Science and Technology Studies ask about expertise and participation? Broadly speaking, research in this area is animated by the evolving contradictions between two prototypical ideas: lay expertise and folk science. Lay expertise posits alternative ways of knowing which are nonetheless crucial for the success of techno-scientific ventures (Wynne 1996). Folk science, on the other hand, investigates how amateurs compete and challenge scientists and engineers in their own playing field. The former is epitomised by Bijker’s phrase that “building water disposal systems and nuclear power plants involves more than what is described in engineers’ handbooks” – the latter by Sismondo’s aphorism that “people sometimes take science into their own hands” (2010, 187). Most research in the area is occupied with some of the implications of both observations. I use Sismondo’s survey to present the most important problematics and then show how they bear on the research questions addressed here.

First, there is work like Ezrahi (1990) which investigates how scientific and technical knowledge is used by the state to justify its actions. Conversely, Söderberg (2013) argues that often hackers themselves use the justification of pure efficiency in order to advance their own political agendas, a justification built up by the state and capital. **Second, there is much research on alternative forms of expertise**. The seminal contribution in this area is Collins and Evans (2007) who introduce the category of interactional expertise. Interactional expertise is the ability to converse on the matters which fall into an area of expertise without necessarily being able to contribute to the field. Later on I propose the idea of potential expertise – the ability to learn or to project the ability that one is able to learn – which plays a crucial role in the dynamics of participation and expertise in the hacking scene due to the auto-didactic ethos. Roughly speaking potential expertise is the expertise of expertise: the know-how of becoming an expert in whatever field one chooses. It has became part and parcel of the stereotypical hacker persona and a basis for the identification and selection of peers.[2]

Third, a large section of STS literature deals with the relationship of the state and its citizens regarding the negotiation and justification of decisions about the deployment of scientific and technological projects. The original inspiration for this line of research is the model of the Danish consensus conferences created by the Board of Technology (Sclove 2000). Consensus conferences bring together citizens with other stakeholders such as experts, government representatives and corporate managers to discuss the social implications of an emerging technology and make recommendations about the proper regulatory measures. Wynne criticises this approach for a narrow focus on decisions which can easily disregard the differences in local cultures and the diverging technological frames of social groups, even though research presented in Einsiedel, Jelsøe, and Breck (2001) suggests that the Danish model can "travel well". In the same manner, I contrast consensus conferences as one model of democratising technology and dealing with the contradictions between expertise and participation below with my results about grassroots research and development practices in hacker cultures. Basically, in my account consensus conferences are "end of the pipe" solutions to the problem of democratising technology.

Fourth, the self-organisation of knowledge production which partly but interestingly relies on the self-knowledge of organisations have received constant attention from STS researchers at least since the 1990s. When concerned laymen and other stakeholders work together in a more sustained way than at consensus conferences, like in the case of muscular dystrophy patients analysed by ("The Growing Engagement of Emergent Concerned Groups in Political and Economic Life: Lessons from the French Association of Neuromuscular Disease Patients" 2008), these "hybrid research collectives" can improve the framing of issues. However, in many other cases like AIDS (Epstein 1996), Sick Building Syndromes (Murphy 2006) or some epidemics (Borburn 2005) and even the Fukushima nuclear disaster (Kera, Rod, and Peterova 2013) citizens have to make their own research if they want their voices heard by those who should take care of them. Citizen science in times and places like these is

[2] Because of the radical openness of the hacker(spaces) scene, interactional expertise plays significant part in the discourse: the most striking point is perhaps the amount of people who engage in passionate debates about cryptography without being able to influence the field through their contributions, or worse, who influence the field through their contributions, even if on closer inspection they turn out to be unaware of the basic principles of cryptography.

literally a matter of life and death.

What is particularly interesting for us in these accounts is that the projects under consideration rely on distributed open collaboration for the production of expert knowledge which is even recognised as such in most cases. In particular Kera, Rod, and Peterova (2013) highlight the role of hackerspaces as local resource centres where critical (in both senses of the word) technological needs of the community can be realised even in the face of denial from the relevant institutions. In a sense DIY Geiger counters are the open hardware equivalents of the modem connections and fax propaganda provided by the Telecomix hacker group to the Egyptian and Syrian popular opposition in times of civil war and under the conditions of information warfare (personal experience). What is somewhat boring about the aforementioned citizen science initiatives is that the production of technological solutions is relegated to a core group while the general public is largely relegated to the location and collection of data points (Benkler and Nissenbaum 2006). Nonetheless, citizen science projects like SETI@home are justifiably cited as principal examples of commons based peer production (Ibid.).

Fifth, when people really take science into their own hands, hackerspaces research begins. The prospect of out-of-band, grassroots technological research and development are truly exciting from the point of view of participation and expertise. It is arguably in the design and prototyping phase where most of the inscription is done with a technology and where both contingency, material agency and historical hegemony exerts their full powers. Therefore it appears vitally important that citizens are able to influence this phase in the life of a particular technology or product line. Hackerspaces are so to say uniquely positioned to intervene at this most sensitive moment. Acting in concert as a veritable network of humans and non-humans, or a hybrid material and immaterial distributed infrastructure, they can radically reconfigure entire technologies. Later sections will deal with the transformation of military technologies such as drones, industrial ones such as 3D printing or state solutions such as wireless authentication roaming which happened in the span of a few years through the efforts of the hackerspace scene. Following these analysis I argue that these are the most advanced examples of the actual democratic development of technologies, since concerned citizens are the initial driving force. Of course this still means that these feeble approaches suffer from severe limitations and exhibit challenges which can only be addressed fully by a popular revolution. There are few studies which address these developments from an STS perspective. Von Hippel's research programme carried out with exemplary persistence over the years is an interesting starting point (von Hippel 1986; von Hippel 2005a; von Hippel 2005b). The main thrust of his investigations is on user driven innovation, animated by the increased role that empowered user communities play in the innovation process.

In his later work he documents and analyses the necessary material conditions for the emergence of user driven innovation. His account of the political economy of cheap programmable microcontrollers on the consumer market is instrumental for my analysis and it will be developed throughout all the three case studies. It reminds us that neither popular participation nor extra-institutional expertise are solely questions of culture. These points complement the highly developed accounts in the peer production literature of communication technologies and their effect on collaborative immaterial production in software and knowledge bases. **The moral of both the hard and the soft version of the story is the same, sharply formulated by Sismondo as "democratised citizen science requires democratic access to resources more broadly" (Sismondo 2010, 188).**

While these perspectives are all relevant to address the main research questions here, it is ultimately necessary to situate expertise and participation within the exact coordinates of the thesis.

The enigma of expertise at the core of this research project is how relevant knowledge can be developed outside of the traditional institutional settings in a way which takes technology development to alternative directions? Institutions and their infrastructures produce and control the vaster part of expertise out there, and much which can be learned outside of them is ultimately originating from amidst their walls. What enables hackers then to engage in alternative forms of knowledge and technology production and how far such attempts have been taken? In that sense the quest for solving the enigma of expertise starts in a rather hopeless situation.

The problem of participation on the other hand is that given the radical openness of the hacker scene in general and hackerspaces in particular (e.g. that anybody can log in or walk through the door at any time): how community members and peer production collaborators are chosen? It is a fact that most people do not try to become a hacker and most people who try fail. However, liberally speaking the road is open to all: auto-didactic resources and community support is initially granted to everybody. It is just that for most people a hackerspace is a boring or even annoying hangout. In light of these considerations it can be stated that at the outset the problem of participation looks an easy one because the conditions for the democratic involvement of the population in the hacker venture are ideal. But surely there are obstacles to face on the way.[3]

Putting the two together, the question relating to peer production is **how to ensure the effectiveness of the results in a highly technical venture carried out on the basis of open collaboration?** What are the particular limits of participation in collaborative open hardware development and the limits

[3] When I say "I write social science on hackers", most people who are not freaks state that "Me, I don't know anything about computers. Your screen looks really complicated." Strange thing is that this mostly happens when I am not implying or questioning my conversation partner's technical skillage. There must be something fundamentally scary about computing which most people respond to.

of technical sophistication that can be achieved? Or to put it simply: do too many cooks spoil the broth?

Of course both effectiveness and sophistication should be understood here in a double way: as a result-oriented quantitative measure and as a normative political quality. Feenberg's distinction between primary and secondary instrumentalisation of technology articulates these differences. The debate over the problems which technologies should address and the criteria for judging their effectiveness is a main point of contention between the various mainstream engineering cultures and hacker development practices. Again, Feenberg's subversive rationalisation thesis identifies the potential playing field here.

4 Methods

The choice of methods and the exact way to mobilise them for a particular research project is a highly strategic affair. It has to do as much with abstract, almost mechanical considerations, as with the contingencies of what we have at hand in terms of empirical materials. Finally, it is an especially political moment since it determines to a large part the truth which we will arrive at. Ultimately, methods should be understood in the context of a research strategy.

In order to answer to these heterogeneous standards, the current chapter is comprised of three parts. First, I describe the main methods used in the research. Next, I elaborate my approach to case studies. This is necessary since there are many ways in which social scientists use case studies in their research, and they involve very different research strategies and methodological foundations. Then, I give a brief account of each case, justifying its inclusion, as well as its place and function in the research project as a whole. In the last section I explain the methods used for data gathering and analysis in more detail.

4.1 Main research methods

The three main research methods used are *object biography*, *technical interrogation* and *critical historiography*. The following sections describe these methods and the rationale for making use of them. The methods applied are not new — indeed, they have been used in various disciplines since a long time, especially in Science and Technology Studies. However, they are seldom discussed under the following headings and distilled into formally distinct research methods. I feel that it is necessary to provide my own conceptualisations in order to distinguish my methods from similar but differing research strategies.

4.1.1 Object biography

What is an *object biography* as a method for research in the social sciences? The goal of an object biography is to start with something that is very open to empirical, ethnographic investigation – a concrete, tangible, singular object. Something that you can take and put on your table in front of you. The next step is to trace its trajectory through its life cycle and record the assemblages, networks and flows in which it participates. This work usually involves field work and participative observation, interviews, and archive work. At this level the research often oozes into the ethnography of infrastructure (Star 2002), although the starting point and primary unit of analysis is a single object. Eventually, it should be possible to recognise the contours of larger social structures and processes in the lines of these inquiries.

In a way one of the foundational texts of anthropology, Malinowski's classic *Argonauts of the Western Pacific* (Malinowski 1922) is nothing more than an *object biography* in its method, appeals to holism notwithstanding. Actually, the first attempts he made at studying the "savages" ethnographically was exactly to "do" technology: "It was easy to look at it and obtain the names of the tools, and even some technical expression about the proceedings, but there the matter ended." Malinowski identifies the study of technology as the easiest form of anthropology. Later, when he gets more integrated into the community life of the village, he documents the circumstances of building the canoes used for the Kula expeditions, which lead to the most extensive study of village life from a holistic perspective. Ultimately, the whole institution of the Kula exchange system is revealed through following the movement of the Kula objects and describing the ways of handling them. Both time and space dimensions are presented through following the movement of Kula objects and documenting their history of exchanges. *Malinowski started to look at concrete objects in their social context and arrived to a convincing account of a social institution.* Here, we have a similar objective regarding *peer production* and the technological artefacts which we treat, for the moment, under the *open hardware* rubric.

Therefore, *object biography* – at least how I make use of it – is **a movement from the concrete to the abstract, from the small to the big, from particularity to totality**. The shapes and points of passage which such movement crosses are obviously reminiscent of a Hegelian Marxist approach, especially the dialectic of immanent and transcendent critique as developed by Theodor W. Adorno (Söderberg 2011, 22; Adorno 1976, 12). However, where Adorno recommends an alternation between the perspective of the totality (transcendence) and an anthropologically emic viewpoint which situates itself in the life world (immanence), *object biography* at its best proceeds by degrees through the connections between the two. As explained in the previous chapter on the theoretical framework, this is possible for two reasons: ontologically or metaphysically, because I postulate a fragile continuity between immanence and transcendence through a Deleuzian or Spinozist expressionism, and methodologically, because I mobilise Actor Network Theory as a tool for building a bridge between particular and general entities on a flattened ontological field. Béla Balázs (1998) calls this jump from immanence to totality a salto mortale. Thus, it is an admittedly idiosyncratic way to realise the dialectics between immanent and transcendent critique.

In line with these initial observations, it is possible to concertise how such method can proceed. First, it is *an ethnographic description of an artefact, ideally from its inception through its usage to its eventual demise.* Such a description should take into account who made the object and why, how the functionality of the object has been worked out and implemented in matter, what traditions of craft or engineering guided the hands of its creators, what accidental properties rose from its form, how it has been presented to its audience, who put it in use and to what uses, if it was banned or on the contrary, promoted by the powers that be, what it means to the people who wielded it and to those who are deprived of it, and how it is superseded by another class of objects in a technological cycle. Secondly, *a description of the assemblages, networks and flows which the* *artefact entered and altered through its course.* This aspect of research Akrich (1992) calls *sociography*, since it is more concerned with the interaction and arrangements of elements than with an account of a single field. What is it compatible or equivalent with? How does it differs from those others? Does it act as a point of transformation in some larger scheme, like some aspect of an ubiquitous infrastructure? Which roles does it play in and outside the market? Kopytoff points out the process of individuation as a particularly enlightening category to ask with here (Kopytoff 1986). Furthermore, objects do not come in individual units, but whole packages and systems and technological paradigms in which each component is only meaningful in relation to the others and the whole. Third, – and this is where it becomes a truly useful tool for the social sciences –, *how the artefact fits into the totality of human and material relations?* At this last point the roots of such a line of inquiry in political economy become clear.

To summarise, writing *object biographies* in terms of the construction of concrete artefacts and their corresponding architectures as structured spaces of possibilities enables a better grasp on the entangled problems of materiality and sociality.

Last but not least, the pragmatic reason for choosing *object biographies* for my case studies is that my subjects hesitate to talk about their individual selves and their personal lifes to researchers, because privacy is one of the central values which grounds the identity of hackers. Conversely, they are more than happy to discuss their technical work, projects and results, since these are the topics which are in the centre of hacker discourse anyway. So when I walk into a hackerspace as an anthropologist, it is much easier to collect data on technological artefacts than on personal attitudes, life histories or social relationships. That is one reason why names and individuals are largely absent from my ethnographic accounts. In fact, as the example in the beginning of this section illustrates, hackers are not that special in this respect – Malinowski had similar experience with the Trobriand islanders, when on his first real field trip he felt that technological enquiry is the easiest route to approach the social life and customs of the village.

An *object biography* aims to grasp the social role of technological artefacts as a historical process, starting from the details and reaching out to wider social connections. It is implemented through largely ethnographic methods, studying social structures and processes in their objectivity. The reason to employ *object biography* for the study of the implementation of peer production in the hackerspaces is that through the concrete objects it is possible to investigate social practices in their materiality. Finally, in the particular field where the research is carried out, technological artefacts play a central role in the discourse, while human sociality, especially the life of individuals, is often intentionally hidden. My research practice follows the cultural geometry of the field, pays attention to its accents and respects its boundaries.

4.1.2 Technical interrogation

Technical interrogation is the method of looking, so to say, at the social functionality of technical functionality, in detail. While the aforementioned approaches are discussing technology in general or a particular technology taken up as a single unit of analysis, *technical interrogation* seeks to decompose its object into a multiplicity of functions, each with its own history and its own way of integrating into the social whole. Of course the reverse is also true, where *technical interrogation* looks at the effects of composition, e.g. how that technology has been assembled from an array of sub-functions. More then with other treatments, this method requires some understanding and sensibility to the technicalities involved in the construction of technological artefacts, as well as a closer anthropological look at their use. This is where I can mobilise my experience as a practitioner.

Such work has been done in the framework of Science and Technology Studies (for instance Latour 1996; Spitulnik 1994), but Code Studies and especially Media Archaeology has made *technical interrogation* its main concern. This could be effectively accomplished through the radicalisation of the research programme behind the McLuhan (Toronto School) inspired media studies, which was always looking at the material effects of media and how it structures its throughput (McLuhan 1964). However, this could not happen because these people were too close to literature to develop a truly technological sensibility. Actually, as Winthrop-Young (2000) points out with great precision, this line of inquiry had to be formulated in a more or less explicit reaction against hermeneutics on its own homeland. Thus, the necessary work was carried out by German philosophers and media scholars, following the tracks of Friedrich Kittler. Building on these results, first those with a background or interest in the plastic arts, like Lev Manovich (painting), then those with a technical background like Alexander Galloway (coding) – and I am sure I am missing the real pioneers like Jeanette Hofmann here (Hofmann 1999) – started to write about technology in the same way that cultural studies was already writing about art and popular culture. Using the emic technical vocabulary and analytical models, they started to ask social scientific questions which put these figurations in their socio-historical context. The result was a socio-technical genre which went well beyond the vague theories of media and communication studies to understand how a piece of technological medium actually works in the context of intersubjective reality.

The experiments of Fuller (2003; 2005) show how far such an approach can go. His starting point is the exercise devised by Donald Knuth [the legendary author of a multi-volume classic on computer algorithms Knuth (1997); Knuth (2011); et cetera]: “Analyse every process that your computer executes a second.” (Knuth 1989) For a social scientist such an exercise leads to the realisation that the “vectors that connect one thing to another, an instruction to an object, a node to another, a layer to a filter, are always political at the same time they are technical and aesthetic”. (Fuller 2003, 103) Fuller later develops such approach to new media derived from software studies under the title of *media ecologies* (2005), and in a closely related current, Jussi Parikka writes on old media using the term *media archaeology* (2002).

Writing in this drift offers three advantages over the stock Science and Technology Studies approach. First, its *paranoid* disposition, in the sense of Derrida – an inquisitive tuning which aims to connect everything across the field, a certain will to knowledge. The result is what Fuller calls “the unfolding of the particular”, which can capture small effects as they travel through scales like a widening crack. Second, its associated affinity with *computer forensics*: a technical affinity which is yet sensitive to functional features as signs and traces of human action. As a hermeneutics of suspicion (Ricoeur 1970, 34), it can discover and evaluate events which users or designers are not necessarily aware of. Third, as both ecology and archaeology suggests, it does not exclude, like Actor Network Theory, totality effects which prevent a *transcendental critique*. My own *architecture* theme fits in with these discursive strategies, providing the three features in a way that blends consistently with the materials of my analysis.

To recap, concretise and extend the definition, ***technical interrogation*** **is the study of a technological artefact in terms of its functional composition, taking into account its parts and the way they work together.** In order to understand how such an artefact is embedded in social practices, it also includes the toolbox mobilised in the everyday manipulation of such an object. Ultimately, it extends to the technological system in the context of which this particular object becomes a meaningful and useful tool. Latour notes that the “question of how many elements compose a technological system cannot be answered by ordinary arithmetic” (1996, 107). The *technical interrogator* thus draws up a bill of materials, the principles of design, the user manual and the possible reasons for obsolescence. These are all read, observed and tried out for the purpose of a double operation: first understanding them in technical terms, and then evaluating their “extra-functional significance” (Marino 2006). Naturally, while the research proceeds in this way, the course of the presentation will be the opposite: the interesting extra-functional notions come first, underpinned by a selection of the relevant technical details.

One of the main conclusions which I should be able to prove conclusively in this thesis by way of technical interrogation for instance is that the toolbox, the design principles, and the functionalities of open source hardware projects include specific elements which prevent the functional closure which is usually discussed in SCOT style Science and Technology Studies under the rubric of stabilisation, and in the Actor Network Theory school as black boxes. In such a reading of open hardware, openness includes doing away with black boxes and building technologies in a way that they cannot be stabilised absolutely. This goes against two misconceptions: one about open hardware in particular and one about the social scientific understanding of technologies in general. On the one hand, open source hardware studied ethnographically as a social practice cannot be reduced to the question of proper licensing, but includes considerations which cut into organisational cultures, design practices and technological systems – the assemble which I intend to catch under the category of *architectures*. On the other hand, technological development is not simply about different social groups trying to impose their own interpretation of the object as such, a process which drifts teleologically towards stabilisation, but a more fundamental struggle which crystallises in the social conflicts around open source. That struggle is exactly about the extent and the way stabilisation should happen, namely, how accessible ("hackable") a technology becomes for different social groups. Of course, questions of participation and expertise come strongly into play at this point. This is why I suggest the concept of *unfinished artefacts and architectures* in the title – terms which hopefully become progressively clearer as the argumentation proceeds.

4.1.3 Critical historiography

In line with the constructivist ethos outlined above, one can argue that in an age where everything is presented as "new", it is a political act in itself to ask for the histories. Beaulieu, Scharnhorst, and Wouters (2007) emphasises the advantages of using historical scholarship in combination with ethnographic methods, especially when approaching technologically driven phenomena, since it complements cross-sectional with longitudinal data. Therefore, I set it out as a methodological principle to approach each unit of analysis – peer production, hackerspaces and open hardware projects — from a historical perspective a well. Of course, according to the Hegelian Marxist theoretical framework, these histories deny that essence is determined by origin. So the essence of a technology is not set in stone in the moment of its conception or creation. Finding the origins is not the point when putting things in a historical perspective. On the contrary, essence is historically conditioned, and can change with use – it is something which can be taken up as the object of work, or phenomenologically speaking, as the object of intentionality.

While such an explanation clarifies what is understood as history, it does not rigorously address what is meant by the critical prefix. In a non-critical historiography, the treatment of the subject matter is nominally restricted to the work which is carried out by the "subjects" of the research, be they relevant social groups as in the case of Social Construction of Technology or even machinery, as in the case of Actor Network Theory. Critical historiography extends this to the work of the observer, so that the observer is also seen as doing work on the technology, which is potentially directed at its essence. Naturally, following such a methodological move, the observer is not much of an observer any more, since she is involved in the co-construction of technology. Once it is accepted that researchers are part of society and not an alien species which come like colonial anthropologist to survey the Blue Planet, the notion that they participate in the co-construction of society and technology should be uncontroversial.

To summarise, **critical historiography considers its units of analysis as interventions in history, and their internal movements as the shaping of their own history.** Therefore, peer production is work on the category of work itself; hackerspaces interrupt the lineage of hacklabs and shared technology workshops, transforming online hacker communities; while open hardware intervenes in the tradition of electronics development. Moreover, considering the embeddedness of these units of analysis, hackerspaces change the face of peer production, while open hardware projects structure the way hackerspaces are equipped and organised. Finally, the proposal of unfinished artefacts and architectures as a theoretical perspective on these processes is designed to intervene in the abovementioned social practices as much as in the work of their interpretation.

In more general terms the notion that historical awareness is beneficial is widely accepted across a wide range of intellectual streams. In critical theory the mission of the critical intellectual is to clarify the history of the concepts through exploring their internal contradictions and thereby infusing class consciousness into the masses. In psychoanalysis, original traumas have to be recovered and reflected existentially as a form of treatment. In existential phenomenology or phenomenological hermeneutics, the quest for authenticity begins with recapitulating the tradition in which we stand as *our own history*.

Acknowledging the relevance of all these perspectives, it can be stated that a **critical historiography does not merely document the facts it finds but also seeks to develop an authentic relationship with them.** In this case this means the political-strategic evaluation of these historical movements in the context of social conflicts. Ultimately, critical historiography is as personal as it is general: in order to realise the full potential of a research project, one has to realise her own position to articulate the relevant truths best accessible from this given position.[4]

4.2 Case study approach

As Stoecker and Yin point out the case study is not simply a data collection, data analysis or research design method but an all-encompassing research strategy or frame (Yin 2002, 14; Stoecker 1991, 98). The problematic of case studies has to be addressed on three consecutive levels. Firstly, the case for applying a case study to answer the research questions have to be made. Secondly, it has to be clarified what type of case(s) are constructed and in which way they are used by the research. Thirdly, it has to be argued why the particular case(s) make a good fit. These answer to the questions **(a)** *"whether a case study approach should be applied?"*; **(b)** *"how is it to be applied?"*; and **(c)** *"to what objects should it be applied to?"*. The following paragraphs take up these questions one by one.

[4] For instance Coleman (2012) is an excellent ethnographic monograph on hackers in the free software movement exactly because it valorises her particular position in the scene as an anthropologist with a deep technical and social understanding of the issues at hand based on years of experience, a woman, and a non-practitioner. Other interpretations of hacker sociability are possible, but the case for this particular interpretation can only be so sound because she wrote it.

The primary concern is if the case study approach fits the research question. The main research question is *"How the model of Commons Based Peer Production is transformed when transplanted to a different domain, namely from free software to open hardware?"* and the main research site are the hackerspaces. Since the question is asking about qualities (the "how"), it is straightforward to answer it using a qualitative approach – and in fact Yin (Yin 2002, 5) specifically recommends the case study strategy when "how" questions are asked, the phenomena is contemporary to the researcher and there is no overwhelming need for behavioural control as with experiments (Yin 2002, 5). However, one may argue that it would be possible to break down qualities into measurable variables and pursue a quantitative research strategy. The problem is that even then, the variables themselves would have to be determined based on an initial qualitative inquiry, since they are not adequately spelled out in the existing literature. Moreover, as Scranton persuasively argues, the case study approach shows the interplay between a greater number of variables, and often finds ones that would be lost in cross-sectional quantitative research (1986, 285–286). Indeed, even strong proponents of quantitative research admit the value of initial qualitative inquires.

Moreover, the research question aims at producing knowledge about the complexity of interaction between three elements: **peer production** as *an emergent paradigm of organising work*, **hackerspaces** as *a particular implementation of peer production* in the context of hacker culture and embedded communities, and finally, **open hardware** as *particularly problematic products*. Case studies are especially suited for the exploration of complex interactions between elements of various levels of abstraction, since the researcher is immersed in a rich environment and able to gather various types of data, triangulating the results.

Finally, in the main research question above, the elements involved in the research are to be understood as social practices. In the extended formulation of the main research question, *"I ask how these particular practices differ from the general model of commons based peer production, especially the well researched free software development model"*. Social practices cannot be studied without ethnographic methods – specifically participative observation – which immerses the researcher in a form of life, and ethnographic results which seek to intervene in the understanding of broad concepts like peer production are best presented in the form of case studies. As I argue presently, case studies have been showed to have distinct advantages not only as aspects of data gathering in particular or as a research in general, but specifically as perhaps the most important genre of writing which propells the relevant sections of the social sciences forward.

Ultimately, however, the reason for employing case studies is more simple and prosaic than the above paragraphs may suggest. Case study is the most popular method in the field of Science and Technology Studies and the one which contributed most to the theoretical development of the field, as well as to its impact on other disciplines. My research aims to contribute primarily to the field of Science and Technology studies and it seeks to address core concerns in that field such as the *co-construction of technology and society* in general, as well as *expertise* and *participation* in particular. Therefore, it seems straighforward to make my contribution in the form of case studies. Indeed, as Beaulieu, Scharnhorst, and Wouters (2007) points out, Science and Technology Studies is built on a number of *emblematic case studies*, (Oost 2003; Latour 1988a; Bijker 1995; Galison 1997; Franklin, Lury, and Stacey specifically) which constitute both the core knowledge of the field and also embody its contribution to related and overlapping fields such as the sociology or media studies. Theories which have been developed in the discipline are closely associated with their relevant case studies, because doing case studies is an important part of theory building in this area.

The case for case studies as epistemic tools in social scientific research in general and Science and Technology Studies in particular is compellingly made by Flyvbjerg (2006). Since the article is clear, concise and well documented, it is sufficient to give a high-level summary here. Flyvbjerg identifies five common prejudices against case studies which are present in the methodological literature of the social sciences, including that they are not useful for theory building and generalisation, they are too subjective, etc. The interesting move in the article is that he uses the very insights – about science as a practice as opposed to science as a theory – which have been produced by Science and Technology Studies and the related fields to counter these five "misunderstandings", as he calls them. Specifically, he identifies the reliance of much of the methodological literature on a scientific ideal which looks at logics and physics as its model. Of course, as it turns out, neither logicians nor physicists work according to such a theoretical model. His principle example is the single (thought) experiment of Galileo, who refuted the Aristotelian conception of gravitation which reigned for around two thousand years.[5]

Therefore, instead of sound logical principles, Flyvbjerg bases his argument on the social psychology of learning, and the constructivist analysis of how scientific results have been produced in the social sciences so far. This is great because he can ground Science and Technology Studies methodology in the well established previous results of the same field. As it turns out, social psychology – along with Bourdieu's theory of practice (1977) – tells us that learning beyond the rule-governed beginner level largely happens through the aggregation of cases which contain contextual as well as conceptual information. Furthermore, the inception, reasoning and justification of theoretical advances in the social sciences also happens largely through case studies. Wyatt and Balmer (2007) go as far as posting the questions "Are (ethnographic) case studies the only way to do STS research?" and "Have case studies became the *de facto* method within STS?". Retrospectively, the above argumentation addresses the possible objection that in fact most methodological questions are already decided in the moment of the inception of the research question, and as demonstrated in this section, follow

[5] Yin (2002), which is the standard reference on the topic of case studies, also points out the similarity between experiments in the hard sciences and case studies in the social sciences (cited by Keddie (2006)).

logically as corollaries.

The secondary concern is the manner of application. Here it is necessary to immediately include a disclaimer. The present research project aims to illuminate social practices in the hacker scene based on many years of active and a few years of scholarly engagement, spanning many contexts and a wide range of sources, observations and data formats. Therefore the material which is presented is not exactly confined to the three explicit cases in question. Following the suggestion of Stoecker, the case studies here serve to establish the *frame* of the research, the window through which we look at the world of peer production in the hacker scene (1991, 98). In fact the case study as a research strategy have been introduced – virtually from the beginning of the research project – exactly to structure and organise the material, both for the purposes of analysis and presentation.

However, as long as miscellaneous comments are helpful for the proper understanding of the phenomena at hand, they are not excluded from either, so that the richness and complexity of the case can be fully appreciated in the finished accounts. To summarise, the most important role of the case study in this particular research project is as a manner of presentation – basically as a story telling device. As the reader becomes familiar with the everyday life and many strange adventures of these items in the inventory, she is gradually and pedagogically introduced to the social milieu which they inhabit and the lessons it teaches us on the larger questions of life with technology – just like in any good science fiction.

Still, what kind of case studies am I talking about? Yin (2002) and Flyvbjerg (2006) propose two taxonomies which help to locate my position between the various ways in which case studies are utilised. Yin (2002) distinguishes three different uses of research methods: *exploratory*, *descriptive* and *explanatory.* On that scale I aim at the *descriptive* end: the goal is to take up some concepts and take them through the research project, only to return them to the community of scholars involved in the particular conversation around them, but return them transformed substantially, hopefully to the better. Since these concepts work on different levels, the results will be articulated on different levels too. **Peer production** as *a grand theory about the social organisation of work* in contemporary capitalism and/or beyond; the **hackerspaces** as *exemplars of the interplay between participation and expertise* in a middle range theory fashion (Merton 1968; Boudon 1991); and **open hardware** as a particular problematic linking the two in a way that *questions the technology/society nexus* so important to Science and Technology Studies. Cases of open hardware projects are therefore built to address all or at least some of these issues.

But how do they address them? The following categories devised by Flyvbjerg (2006) are useful here: he lists a handful of possibilities in which case studies can be useful for theory building and generalisation (230). Of these, *paradigmatic* and *critical* are picked up here. *Paradigmatic cases* "seek to develop a metaphor or establish a school for the domain that the case concerns." When I propose *unfinished artefacts and architectures*, it is exactly what I aim at by presenting three case studies which illuminate a particular way in which peer production can be implemented. I show that peer production as an unfinished architecture can be understood as *work embedded in the richer fabric of life* – whether this is good or bad – and open hardware as unfinished artefacts is not a licensing question but a concrete set of social and design practices.

At the same time, the case studies should also serve as *critical cases* in the sense that if hackerspaces as embodied and localised communities of practice and open hardware projects as the production of physical goods are considered a form of commons based peer production then peer production itself should not be tied to the ideal type of intermediated mass collaboration of intellectual goods like the Linux Kernel or Wikipedia in the way in which it is discussed in the works of Benkler (2006) for instance. While such a debate on the generality of peer production is largely over in the literature – with overwhelming evidence piled up on the side of a generally positive conclusion –, the current frontier is to understand how the peer production model is reformulated in different contexts. Since the open hardware projects in fact ubiquitously contain an element of both software and hardware design, they are liminal objects where the converging and diverging tendencies in these two domains can be studied in their complexity.

Finally, the tertiary concern is the selection of concrete cases. As the SAGE Dictionary of Social Science Methods states, "a key factor affecting the succes of the study will be the criteria for the selection of the cases to be studied" (Keddie 2006, 20). As stated above, all three case studies included in the thesis are conceptualised as paradigmatic and/or critical in some ways, and it will have to be shown how. The next paragraphs present each case one by one, explaining the technological artefact chosen as the subject of the case study, outlining how is it paradigmatic (or critical) and show how it clings together with the other case studies.

4.3 Justification and presentation of cases

4.3.1 Case 1: The r0ket device

The first case is the *r0ket*, which is basically a geeky name tag for geeky conferences that doubles as a pioneering programmable ARM microcontroller development board. It is also called rapid prototyping board after the famous Arduino which became a poster-item of the scene – see Banzi (2008) for a user friendly technical and Paoli and Storni (2011) for a Science and Technology Studies friendly sociological introduction to the idea of microcontrollers in general and the Arduino in particular. Both are good introductions for the other two case studies as well, since they include similar technological elements, technological systems, and design concepts.

The r0ket is a *paradigmatic* case for an open hardware project undertaken in the hackerspaces milieu using commons based peer production because it has been devised and built in a particular hackerspace (µC3 in München, Germany), distributed in emblematic hacker meetings (Chaos Communication Camp and Chaos Communication Congress in 2011), and generally aimed at nothing more or less than the rich embodiment of hacker ideals. I claim that since the authors of r0ket sought to make this technological artefact as appealing as possible to the general hackerspace audience, they have already undertaken most of the work required by the ethnographer – that is, the gathering of cultural traits which make the participants of a certain subculture *tick*. Thus the r0ket device belongs so closely to the hacker scene that it has virtually no meaning or application outside of it. It is a self-enclosed artefact which contains the most important vectors of its cultural context in a nutshell. Its positive reception in the community proves that it achieved its goal, although it did not caught on as part of the typical development tool-chain and ultimately fall out of use.

Moreover, it is a *critical case* of electronic hardware peer production by embodied communities since it has been largely implemented in a particular location through physical work sessions, distributed hand-to-hand at physical hacker conventions, and often used standalone or with other electronic components for physical computing projects (on the latter see Igoe and O'Sullivan (2004)). In sum, its design, production, and distribution was as much tied to physical space as much it is possible in the confines of today's ubiquitious computing and the hacker space milieu. Therefore, it can be considered a *"hard"* case of commons based production of electronic hardware. Consequently, if commons based peer production can be implemented successfully in this case then it is likely to work in other cases, and in general commons based peer production applies to physical electronics. Of course as I stated before, the key question is not if this could work but how is the model transformed by transplanting it to other environments which are different from the ones in which the theory has been developed.

To summarise, the r0ket case study shows the development and distribution of a classic, complete, one-piece electronic artefacts in the hackerspaces, describing how peer production is implemented in such an environment and what are the particular difficulties associated with the peer production of hardware. I do not know about any case studies of peer production which would provide an ethnographic and biographical account of a hardware piece, except my Master's thesis (Maxigas 2012b). Here I integrate that ethnographic account in a more advanced theoretical framework and confront it with the conclusions of other cases – the result is a paradigmatic example of the unfinished artefact (the r0ket device) and the unfinished architecture (the hackerspace), as well as a critical instance of open hardware production in embodied communities.

4.3.2 Case 2: Door systems

The second case are the *door systems* installed in a vast number of hackerspaces. A door system is a button next to the door of the hackerspace which you press when you are the first person to enter or the last to leave. It typically changes an image on the website of the hackerspace which shows if it is open or closed at the moment. This is useful because hackerspaces usually do not have fixed opening hours, being volunteer operated clubs which are open to the public but where only members have keys. Most hackerspaces have unofficial members who are not key holders, or official members who have no key for some other reason, and of course many casual visitors too. In order to coordinate all these people without forcing key holding members to open the space in a regular schedule, door systems provide a dynamic solutions which simply lets everybody know when somebody opened the hackerspace. Going beyond this basic functionality, real life door systems usually provide a rich and ever evolving array of functionality, such as dedicated iPhone apps (like in the H.A.C.K. in Budapest), automated interactions with the alarm system (like in BitLair in Amersfoort), or announcing newcomers in a robot voice through a speaker in the physical space (like in the London Hackspace).

What makes door systems a potentially *paradigmatic* case of the unfinished artefact is that there is no formal definition of what a door system is, no single reference design, no central repository for documentation and not even a dedicated user community. It is simply what hackers do in the hackerspaces: a social practice. While the r0ket device is an emic concept which designates a concrete product, complete with a dedicated website (at http://r0ket.badge.events.ccc.de/) which includes documentation and contact details for support, "door system" is my etic term for referring to a number of completely different "hardware hacks" which nonetheless all provide the functionality described above. Therefore, it shows very explicitly – as a *critical* case should – that open hardware is not the question of proper licensing or a conformance to this or that formal definition, but a set of social practices around working with hardware which operated according to the logic of commons based peer production. This is where ethnographic case studies can really shine, since such phenomena can only be captured by sustained field work which identifies cultural patterns "in situ", in the field site. To summarise: I argue that door systems are a paradigmatic case of unfinished artefacts and a critical – one would say corner case in the technological jargon – of open hardware. That's why, after exploring the idea of open hardware and presenting the concept of unfinished artefacts in the first case study, the second case study should be the appropriate site to confront them with each other, and see how I can improve the understanding of peer production through my conceptualisation.

Moreover, door systems are also a paradigmatic case for the co-production of technology and society – or what can be termed *techno-sociality*. In this sense the door systems are a case of a social innovation which is implemented in physics and mathematics and works directly on the social dynamics of the respective communities. Door systems subvert, invert and reinvent the extremely entrenched social institution of the opening time – so prevalent that even anti-systemic anarchist communities stick to it –, and does it through the implementation of technological system rather than the formal discussion, setting and enforcement of community social norms. It is a school book example of *governance through technology*, or employing the more theoretical language defined in the previous sections, *governance through encoding.* Interestingly, this conforms to the strong desire of hackerspace community members to have less explicit social rules and more shiny technological gadgets. As one hackerspace "constitution" states: "Rule zero is that you behave in a way that we should not have to invent new rules."[6] To recapitulate, door systems are a potentially paradigmatic cases of open hardware projects *as* social practices; critical cases for the organisation of peer production through shared culture and social norms; and once again, paradigmatic cases of the co-construction of technology and society. Whereas the first case study reflects a highly localised (territorialised) instance, this second case explores how localised practices spread across the subcultural landscape in a community of practice.

4.3.3 Closing comments

Many would object that the cases of these two small scale electronic artefacts are not interesting because they are not famous, big, successful, and they are not even mentioned in the headlines of news articles. Such criticism does not deserve a lengthy reply since it is clear to anybody familiar with anthropological literature that well chosen examples analysed in the complexity of their embeddedness to local contexts - think of the Balinese cockfight in Geertz (1973), an anthropological classic - contributes more to theory building and generalisation than singling out spectacular exceptions.

Mistaking the visibility, success and mass appeal of a project for theoretical significance, scholarly relevance or the rationale for critical attention is a mistake, often made by researchers who have spent limited time studying the field and therefore only covered the "tip of the iceberg" – maybe down to scratching the surface, so to say.

Given the complexity and relative speed of the problem domain, case studies related to Information and Communication Technologies often fall into this trap, and it is perhaps even harder to avoid it in the peer production literature. Papers and especially presentations whose impact is based on the appeal of technologies (e.g. the empirical material) and not the brilliance of the social scientific analysis can be called "sweet technology". Such slides can be persuasive but only to the point of brochures about cars or washing machines. They seek to mobilise the audience with pictures of exotic projects and anecdotes with a moral, promising you a beautiful life once you hopped on the bandwagon of the suggested concepts.

[6] This in-joke plays on the fact that in many (but not all!) programming languages, counters start from zero rather than one – a common caveat for novice programmers. It is surely an example of how communities of practice reproduce their identity by referring to the tricks of the trade in their parlance, and probably an example of how they sustain their shared situated knowledge. Of course it is also a reference (or self-reference) to self-referentiality – which is known as recursion in programming. It is not by chance that Kelty (2008) refers to free software hackers as a *recursive public...*

The rhetorics of success thus confuses the academic radar, which can only pick up the characteristics of the "top 1%" of peer products which "stand in" for the phenomena in academic discourse while the "99%", largely comprised of personal projects, failed initiatives and abandoned efforts, flies under the radar. If one reads through the literature on free software development written around the turn of the millennia, caught up in the euphoria of the dot-com bubble (Raymond 1999; Himanen 2001), the overall impression is that open sourcing a project makes it automagically successful.[7]

Still today, the discussion on peer production revolves around Linux and Wikipedia as the two flagship examples, so that peer production as a model looks like the recipee for success. This is bad for the movement because constructive criticism is lost in marketing talk and therefore the sharpening of concepts becomes almost impossible. It is bad for science since successive projects are effectively overrepresented in the sample and then the big conclusion from the data analysis is that peer production is successful and growing – an obvious methodological mistake. One of the strategic reasons for proposing the term *unfinished artefacts* is to reorient research towards the darker side of the peer production economy and counterbalance such tendencies.

On a final note, I hope it is clear from the exposition above that the two cases are included in the study for what they bring to the discussion and not for the sake of a comparative generalisation in the logical sense. They are definitely there to bring out some systematic features in the peer production and socialised use of open hardware or unfinished artefacts in the hackerspaces scene, but the goal is to represent the diversity of the field rather than a statistically inspired research strategy – although there are obvious connections to to *exhaustive sampling*. Ultimately, the case studies should pave the way of building an overarching argument bit by bit by cutting through the winding roads of on-the-ground complexities and the fog of theory, if such a mixed metaphor is permitted.

4.4 Data collection

4.4.1 Field work

One of the greatest advantages of studying hackers in hackerspaces is that traditional ethnographic methods can be mobilised in the collection of the data, in contrast to many studies which have to rely on the Internet as the only way to observe the research subjects, or where hackers are for instance interviewed but cannot be observed while involved in actual social practices. I claim that despite advances in methodology, digital and virtual ethnographies have fundamental epistemological limitations, while embodied communities can be studied by the well established methods of field work and participative observation. As I have argued earlier

[7] "Automagically" is hacker lingo for a thing that appears to work, but we don't know why. Since hackers like to understand how things work, this is often seen as disturbing.

(in the section on the *Main research methods*), this is crucial if not absolutely necessary for analysing phenomena as *social and technical practices*, which are different from discourse – the object of discourse analysis – or variables as they are treated in statistical methods. These don't mean that either of these methods would not be valueable for knowledge production or developing social theory. They are just less suitable to answer specific research questions.

Of course this does not mean that the ethnographer who sets out to study hackers as they are in their bodies can live without methods associated with the younger and more risky strands of digital anthropology, especially the parts devised for the study of online communities, since social life in hackerspaces have a strong digitally mediated aspect. It is important to keep in mind that "although ethnographers will always be concerned with understanding micro-histories of people in their own space and time, this space and time is no longer seen as fixed and unitary. Instead, it is permeated by the diverse times and 'non-places' of wider, global interactions." (Quoted in Dicks et al. 2005, 117) Hackers may be sitting around tables chatting, but many would keep checking their laptops and possibly even involve others from the chat room of the space in the conversation, or participate in several conversations – online and offline – at the same time. This is not that different from today's academic conferences permeated by mobile media.

During the span of the research project (starting in 2011) I visited many hacker events and hackerspaces with the explicit intention to answer my research questions. I call these shorter visits *field surveys*. Such field missions usually have three objectives.

Firstly, there is usually **a clearly defined primary objective which has to be achieved no matter what.** For instance in 2011 at the Chaos Communication Camp and Congress the primary objective was to conduct participative observation and semi-structured interviews on the r0ket device with the r0ket team and the users of the artefact, because it is one of the case studies and it was released at these two events. The results are summarised in my Master's Thesis. Similarly, I travelled to the Toulouse Hackerspace Festival (2013) to look for good case studies on open hardware – which turned out to be a failure.

Secondly, the usual procedure of field work applies, so that **one keeps hanging around and picking up details or conversations which can be illuminating for any of the research questions on the table.** These find ways into field note files tagged according to the topics they touch, so that fragments can be pulled together later for data analysis. For instance the germ of my conclusions on open hardware stem from an encounter with Basque hackers during the 2012 hackmeeting in Calafou. When I asked them about the open hardware projects they are doing, they did not refer to any concrete, completed technological objects created by them which could be easily reproduced because they are open hardware. Instead, I understood that for them doing open hardware was not about open licences and good documentation, not even about publishing the details of designs on the Internet, but about actively teaching people the skills which are necessary to make their own hardware and to share informally the details of one's own inventions. Hence open hardware for them is not the property of a technological artefact but a social relation, a particular social practice which hackers *do*.

Thirdly, these experiences formed a body of **general knowledge about the scene which enabled me to identify the typical and atypical properties of each case study**, to see how an exception to a rule can be illuminating (in the sense of critical case studies) or how a certain trait can be more descriptive of the general dynamics then another (in the sense of paradigmatic case studies). For example the r0ket stood out as a product of exceptional quality from the host of other projects which I encountered in the hackerspaces, which would suggest that it is not the best candidate for characterising the social practices around technology which are prevalent in this milieu in general. However, looking at its reception in the scene by participants of varied backgrounds convinced me that if so many of my diverse subjects like it so much, it must have some qualities which are generally valued by them about hardware projects. Table 1 lists the dates and locations of these *field surveys* together with the actual event or hackerspace studied.

Table 1: Field missions

Date	Location	Occasion
2010, June	Bratislava	Opening Party of Progressbar HS
2011, August	Vienna	Field visit to Metalab HS
2011, August	Finowfurt	Chaos Communication Camp
2011, December	Berlin	Chaos Communication Congress (28C3)
2011, December	London	London Hackspace HS
2012, March	Amsterdam	TechInc., LAG, BitLair, Hack42 HSs
2012, December	Hamburg	Chaos Communication Congress (29C3)
2012, October	Calafou	Hackmeeting
2013, May	Toulouse	Toulouse Hackerspace Festival
2013, June	Dublin	TOG hackerspace
2013, August	Heerhugowaard	Observe, Hack, Make hacker camp
2013, December	Hamburg	Chaos Communication Congress (30C3)

The other type of field work was more extended – 1 to 3 months at any one time and 3 month total for any particular case study – which gave me the opportunity to have a more deep and grounded understanding of the particular location, the participants and the projects. The emphasis here was more on the observation of processes as they unfold, their trajectories and their embeddedness in daily life or in rites of passage. The tripartite list of objectives outlined above apply to these *extended field work periods* too.

Table 2: Extended field work

Date	Location	Occasion
2011	Budapest	Case study: r0ket
2012, December	Amsterdam	Case study: Door systems
2013, December	Amsterdam	Case study: Door systems
2013	Calafou	Studying participation/expertise
2013	Budapest	Case study: Door systems

4.4.2 Documentation of objects

Before the thesis work, I have completed a single *object biography* on the r0ket device in the framework of my MA thesis. The experience of that work lead me to try the same on something else the *door systems*. The practical process of compiling the data for these object biographies went more or less as follows.

First, I read the available online documentation and textual discussions on the topic and its context. Then I looked up these artefacts where they exist, became familiar with them through usage and experimentation: breaking them, taking them apart and putting them together differently. I observed their producers, fans and users, as well as any of their enemies, if found (in the Pi case there were plenty). While I did conduct semi-structured interviews where I found it appropriate, most of the data was pieced together from casual conversations, where I could easily slip in more directed questions if a data point was missing. That is why my field notes include many “facts”, known, to-be-found-out, to-be-confirmed, and unknown, rather than lengthy quotations and excerpts. “Hanging out” as an anthropological technique was especially helpful here, both for gathering data and for compiling field notes on site.

My experience in this field work has been that the essential task to undertake on a field survey or field work is to establish a working relationship with informers and understand the context and the main outlines of the phenomena – details can be filled out later over techno-mediated communication channels, especially since hackers are easily reachable online. Indeed, mixing the three forms of presence – being at the hackerspace, meeting the same people and things during hacker conventions, and using the Internet to connect – allowed me to follow processes through an extended period.

For instance I was able to visit TOG in Dublin in conjunction with the door systems case, propose changes which I suspected would affect the social dynamics, work online to help implementing them technically, and return later to observe the effects. Such a procedure, spanning over a year, supplied me with much information on the production process, the social context and the technical details too – in line with the three part methodology laid out above, wherein I combine the biography of object with hands–on technical understanding and a historical perspective.

During field work of any kind I kept a digital diary and wrote my observations up to field notes daily.[8] Ideas have been discussed both with the practitioners and fellow researchers, probed and modified accordingly – or simply discarded and forgotten as a mistake. Smaller blocks of a coherent theoretical ideas or historical observations have been flashed out into blog posts and drafts, continuing the recursive relationship with my dual audience, and ultimately worked into the thesis text.

Of course during field work I tried to make it clear to my subjects that I am doing research on hackerspaces in general and the specific artefacts in particular. Sometimes I was dressing in my "colonial outfit" of white linen complete with panama hat, which made me instantly recognisable as an anthropologist in the black clad crowd of baggy trousered "natives". Other times I adopted the latter dress code but wore a T-shirt with a "Field Testing Social Theory" graphics designed for my research. Both strategies were effective in triggering conversations which could begin with the declaration of my affiliation and continue with explicit research questions. Ultimately, I could always justify my presence citing my relevant technical works and go on from there.

4.4.3 Interviews

The careful reader may notice that references to individuals and to interviews in particular are largely missing from my account. There are no transcriptions in the appendix and the text is not littered with literal quotes.[9] While I acknowledge that this is a weak point of the text, I would like to point out a number of reasons for such obscurity.

One is to go along with the culture of paranoia which pervades some parts of the hacker scene – to *anonymise* the text. I dwell on the methodological technicalities of this soon. Another is that I did not include semi-structured and unstructured interviews in the research design as a tool separate from field work. Therefore, interview material is tightly integrated with field observations and desktop research results in the text. In fact I was trying hard to follow the journalistic dictum of not publishing anything unless I have two independent sources. **The last reason is a theoretical bias towards studying observed practices and the commonplaces in cultures, and especially how the two comes together in situations and processes, in contrast to discourse and individual motivations**, for example.

Ultimately, even though my interviewees are hard to identify, the research sites and technological artefacts are well documented, and any researcher can check most facts included in the analyses by consulting the websites and asking questions on the chat channels of the respective groups.

[8] I used org-mode (Dominik and others 2010) to organise the field notes

[9] In fact I was not recording any interviews, did not transcribe them and never thought of using Atlas.ti to perform any discourse analysis – even less so to reconstruct the conceptual universe of my subjects through connections between frequently mentioned terms and topics.

That said, I conducted at least a dozen semistructured interviews for each case study and around another three dozen for the chapter on the research sites, the hackerspaces. In general I followed the guidance of Bernard and Babbie (2006, 210–250; 2010, 318–321, 275–278) in terms of interview design and administration. The actual manner of conducting interviews was the following, in terms of its derivations from the standards layed out in the literature.

Since I felt that given the discourses on privacy permeating the scene, audio and especially video recording would alter my data significantly and make my subjects feel more insecure. Additionally, as Kitzinger and Barbour (1999) point out, video recordings which are usually praised for the richness of the data they capture, can give a misleading illusion of comprehensiveness. Furthermore, I believe that hoarding a growing archive tied to a research project which is dragging on for years can have dangerous consequences for both the project and the researcher. The pressure of the archive, even if organised meticulously, weights down heavily on the shoulder of the ethnographer. Therefore, since it is entirely uncontroversial – on the contrary, very proper indeed –, to sit in a hackerspace with a laptop on your lap, I used the skill of taking the minutes, built up during my activist work to capture field data in typescripts. While this allowed the recording of my impressions about how my subjects exist in their world, it also allowed for instance copy pasting links to relevant information in the archive.

Hackers believe in plain text as the universal interface, and indeed, this allowed me to instantly store the information in a structured, easily editable and trivially searchable form. The same qualities of usability and maintainabily can only be achieved with multimedia content using specialised technology and an enourmous amount of time, and the resulting archive will have to be transcoded every six or twelve years due to the rapid changes in software and hardware. In contrast, there is virtually nothing that cannot be done to plain text using the standard Unix tools installed by default on Linuxen, OS X or even Android phones. These tools have been available since the early 1970s and thanks to their continued usefulness there is no reason to think that they will disappear in the foreseeable future. Of course, such arguments also risk the above mentioned illusion of comprehensiveness.

Moreover, as mentioned above, such data capture strategy allowed me to basically anonymise my interview material on the spot. Babbie (2010, 67) emphasises that social researchers often confuse anonymity with confidentiality. Anonymity means that the researcher cannot easily trace back the information to its source, i.e. the particular person who served as research subject. Confidentiality means that the researcher is well aware of those details but unwilling to share it with others – for instance the authorities. Since many hackers feel strongly about data protection, they are often more sensitive to such nuances then anthropologists. I suspect that there is also a correlation between the higher number of court cases per capita within the hacker population as compared to anthropoplogists and the former's increased awareness about data protection.

Therefore, my data collection strategy was to strive for anonymising at least my records, if not my memory, as early as possible. I did all recording, processing and archiving on my own infrastructure – essentially my laptop which was installed with full disk encryption and my external hard drives for backups where I used encrypted virtual filesystems. Of course this is textbook example of transfering agency from humans to machinery, encoding social relations into physics and mathematics. Methodologically, however, such protocol provides a fair amount of anonymity *and* confidentiality to the research subjects.[10]

4.4.4 Focus group discussions

Finally, the last data collection method tightly integrated with field work is inspired by ethnographic focus group interviews (described concisely by Babbie 2010, 322–324). I adopted my methods to the particular properties of subject matter. The divergences and affinities with the text-book version of focus groups interviewing are layed out in this section, starting from putting the method in the context of the specific field sites. **While hackerspaces give a stable space for their members to interact, hacker conventions provide a fixed time for them to meet.** Therefore, many participants call them temporary hackerspaces, and they have an important role in maintaining the shared discourse, hammering out the direction of the next years' work and negotiating meanings.

Thus, like focus groups, hacker conventions bring together a self-selected sample of participants from various localities matching a specific profile. They traditionally feature a mix of presentations, workshops and informal social events. The first of these is a good opportunity to present one's research to her subjects and engage in dialogue about the findings, because results have to be first verified and developed in collaboration with participants and then by the scientific community. In their programmatic article on the the challenges and promises of focus group interviews, Kitzinger and Barbour (1999) identify two decisive factors in conducting a fruitful focus group interview. One is the crucial role of the stimulus and the other is that of the environment (site and setting), which also greatly influences the results. I believe that in the situation described here both can be set in a way that the researcher has good control over the stimulus while the site and setting are confortable and natural to the interviewees, as required by Green and Hart (1999). Since presentations end with a question and answer section, it is possible to turn them around and ask questions of

[10] There are two exceptions. Firstly, where my informants became collaborators in the research so much so that I felt that their contributions should be credited, I asked them explicitly how they want to appear in the text. Secondly, when I am not using interviews but materials from public mailing lists, websites or chat channels, I usually quote text verbatim. In the former case I acknowledge that the contributor has the right to decide about their own visibility, while in the latter case I presume that this decision has already been taken when the information was inserted into the public record. In both cases I avoid taking the responsibility by delegating it to my subjects, who I understand are resonsible adults with at least some relevant expertise.

the attendant participants themselves, rather than answering questions and comments.

One example of conducting such an informal focus group was during the Iberian Hackmeeting in 2012 at Calafou, where I presented (Maxigas 2012c) my historical research based on an article (Maxigas 2012a). It was a perfect occassion to contextualise the results geographically and gather much new data on two points. On the one hand, attendants were passionate to explain the reasons behind the precursors of the hackerspaces, the politically oriented hacklabs, falling out of use and becoming increasingly irrelevant. Many participated in ones around Spain and Italy and they were eager to discuss their experiences. This born directly upon my analysis of the trajectory of these shared machine workshops. On the other hand, the second point of contestation had to do with the differences between Northern and Southern European hacker culture, taken up in its social-historical dimensions. This discussion grounded my construction of the field site — North European hackerspaces – based on the differences in social geography.

The presentation-turned-focus-group-interview was followed up with semi-structured interviews about particular hacklabs with the most active discussants. I repeated a similar pattern at various other occassions during my field work. Of course such a data gathering tactic is highly precarious and greatly depends on the abilities of the presenter to direct the discussion as well as the occassion and attendance. I found that my experience in moderating assemblies from my activist work was highly instrumental for succeeding there.

The three most significant differences between my practice and the canonical method of focus group interviews are the following. Once, *participants are self-selected and the sampling is more quasi-exhaustive than anything else.* Twice, *as the leader of the discussion I am highly visible and far from impartial.* This second point should not be a problem even in traditional terms since the beginning of the 1990s it is normal for the researcher to appear as a personality rather than as an "objective nonentity" (Kitzinger and Barbour 1999, 14). Thrice, while topical focus group discussions were sometimes repeated as many as three times – the above example was complemented by discussion over dinner with participants of a major Italian radical technology group who participated in the hacklabs of their country –, the major difference to the focus group interviews described in text books is that the focus group discussions I did *could not be reproduced in a similar setting a significantly high number of times.* Therefore, I claim that it fares very well as a data gathering method in the context of ethnographic work, but of course it does not stand up to the statistical standards and enchanced reliability (Babbie 2010, 153–55) associated with the original focus group interview method.

As the title of the section says, the variation of the *focus group interview* outlined here I would rather call *focus group discussion*, or even semi-spontaneous or ethnographic focus group discussion. In fact GreenHart1999a also makes such a distinction between focus group interviews and focus group discussions, stating that “discussions groups bring together peers, ideally participants who have relationships which pre-exist the research setting. The findings from the study on which this chapter is based suggest that such groups can provide data that are useful for health promotion professionals as well as for social theory development” (21). Here, the research setting is an extension of the field setting, thus the tentative prefixes I proposed above.

4.5 Summary

In this section I layed out the methodological foundations of the research, focusing on the methods used and the manner in which they were utilised. The research design is based on a *case study approach*, revolving around 3 cases – which I have justified in detail. The concrete methods I use to construct these cases are *object biography*, *technical interrogation* and *critical historiography.* While I do have my own particular twist on these methods, they basically comprise the standard toolbox of Science and Technology Studies. Finally, the last section gives a more down-to-earth account of the data collection process, explaining where, when, and how it happened, in dialogue with the relevant methodological literature. While the emphasis is on the field work, object documentation, interviews and focus group discussions – which have been integrated into the former – also get their own sections.

Ultimately, what I tried to convey in this chapter is an attempt to adapt and integrate social science methodology to the cultural norms prevalent in my field site rather than trying to tweak the normative tuning of my subjects to accept the intrusion of the traditional methods into their social life. The result is building a more symmetric relationship with the people that I study that enhances participation, leading to higher internal validation of data, without disclaiming my specific scientific expertise as a specialist in the social sciences. The difficulty of such an approach lies in negotiating the tradeoff between decreased reproducibility and increased rapport, or in a more abstract sense, validity and reliability (as seen in Babbie (2010), 155).

5 Free, Libre and Open Source Software

Benkler (2006) defines **commons based peer production** as a form of social-economic production which can potentially involve a high number and a wide range of participants and actors who can work effectively towards a chosen complex goal. Coordination is usually characterised by three factors: (A) heavy reliance on the Internet, (B) lack of monetary compensation, (C) lack of a chain-of-command which is customary in hierarchical organisations. In order to achieve this, CBPP projects have to be (a) *modular*, which means that complex problems can be broken down into a number of more simple tasks; (b) *granular*, which means opportunities for participants with a wide range of motivations to get involved in their own way; and (c) *easily integrated*, which means that contributions can be easily brought together to form a functional whole. Canonical examples are the GNU/Linux operating system and Wikipedia, the Free Encyclopedia.

5.1 Software versus Hardware

Perhaps the most precise scholarly exegesis of the methodology, main historical moments and issues of the free software development method is Weber (2004). Primarily based on that material, Benkler's comments and my personal experience in the free software world, it is possible to draw up a sketch of how free software is developed as a form of commons based peer production. During the presentation I use the factors outlined by Benkler as a scaffolding, fleshing them out with the specifics of free software development according to its technical, methodological, organisational and cultural conditions.

Unlike other expositions on the topic I will give more technical details because when it comes to a comparison between software development and hardware development — the topic of this thesis —, it will be important to understand what is specific and what is general between these two activities. Additionally, I believe that the analysis of any production process — or even any social phenomena — should start with the concrete "material conditions". In other words, the approach taken here is aimed to enrich the Science and Technology Studies perspectives with methodological clues taken from the young Software Studies field (as exemplified by Fuller 2003). Analogically, this could potentially develop into a Hardware Studies research practice. Therefore, the question which will be tackled later is not only if free software development can be extended to open hardware development, but also if Software Studies could be extended to hardware. The advancement of the methodology is following here the advancement of the subject matter.

It is worth to note that this section does not aim to present original contributions, only to critically describe the present state of knowledge which can be mobilised in later chapters as a background against which issues in the case studies of open hardware development stand out. In this capacity it also includes many pedagogic simplifications which step over the complexity of certain issues, yet there should be enough texture to have a feel of what is going on.

5.1.1 Reliance on computers and the Internet

There is a saying that the programmer is a machine for turning caffein and pizza into source code. The core of truth in this saying is that the software industry needs much less initial investment in fixed capital then other sectors of the economy. Information and communication technologies — the most important and obvious of which are the personal computer and the Internet — have became part and parcel of most middle class households in developed economies by the turn of the millennium. As far as fixed capital goes, free software development relies mostly on these consumer-grade products as a means of production.

The versatility and ubiquity of computers stems from the fact that they are general purpose information processing devices. As Cory Doctorow pointed out recently (Doctorow 2012), this is not to be taken for granted at a historical moment when the proliferation of tablets, e-readers, smartphones and other gadgets means that there are more and more devices on the market which are severely restricted in their capacity for general information processing. These can be hardware restrictions like the lack of a physical keyboard or software restrictions like Digital Rights Management (DRM — more precisely called Digital Restrictions Management, see Stajano (2003)) systems which seek to enforce intellectual property regimes on the operating system level.

Of course the most important production equipment for a programmer are software components which are highly specialised. These form an assemblage which is usually called (with little precision) a toolchain. The principal components of a toolchain are the *editor* for writing software code, and the *compiler* or *interpreter* for turning the source code into an executable program. Editors can be traditional text editors like emacs and vi (both from 1976) or even Kate (from 2000), which focus on working with the source code as a form of textual expression, or more sophisticated graphical Integrated Development Environments (IDEs) like Eclipse, Netbeans, or Geany which are often geared towards providing all the tools to work with a programming language (described below) in the framework of a single application.

Compilers turn source code into executable binaries, while interpreters compile source code “on the fly” and execute it at once. As a result, compiled programs can be executed without the compiler, but in order to run interpreted programs the given system has to have the interpreter installed. There is a difference between low-level compiled languages like C, which cannot be run through an interpreter, and high-level interpreted languaged like PHP, which cannot be compiled. Of course the real situation is much more confusing since many languages like Python or Common Lisp can be compiled *or* interpreted, and some others like Java and .net are compiled to their respective virtual machines rather than to native machine code. Other development tools include debuggers, steppers, profilers and so on — mostly diagnostic tools which give information about what is happening during the execution of a piece of code. Finally, it is worth to mention linkers (and loaders) which put together the executable file from the compiled sources and the provided libraries.

If the toolchain is part of the means of production, then *libraries* are part of the raw materials of programming. They are a collection of functions for solving a set of specific problems. For example the PyMongo library allows programs being written in the Python programming language to manipulate MongoDB databases. Programmers can load the PyMongo library and call the functions in this library to perform various tasks like creating a new database and filling it with data, making various queries, and so on. Libraries are a form of abstraction which hide the underlying complexity of the task (in this case, handling the database), only exposing a simple interface to the programmer in the form of a set of documented functions. We will deal with libraries later, in the section on Modularity (a.).

The *GNU toolchain*, whose beginning can be traced back to the beginnings of free software and the pioneering GNU project, is probably the most popular amongst programmers of all kinds. It is free software and used on almost all platforms from mainframes through personal computers to embedded systems. It is licenced under the General Public Licence (the GPL). The GNU toolchain includes all the tools mentioned above except the editor or IDE. While GNU/Linux is usually featured in the academic literature as the standard example of a free software product which became industry standard and indeed, industry leader, the GNU C compiler (now simply called GNU compiler) is another example which acquired the same status much earlier than GNU/Linux. The hegemony of GCC also means that (free and non-free software) programmers in all languages, not just in C, nowadays expect at least the toolchain to be available for free.

If computers belong more to the sphere of Information Technologies, the *Internet* is arguably more related to Communication Technologies. In the beginnings of computing programming was much closer to the arts and crafts in the sense that the associated knowledge and skills were attached to specific persons. When DEC was selling mainframes like the early PDP-11s, the installation of the operating system was done on-site for each computer, so that programmers had to travel from the company or other university departments to set up the machine. In the early days of personal computing programmers came together at specific places like the Homebrew Computer Club or at specific times like the Demo Scene parties to exchange casette tapes and floppy disks, and in Hungary, pirated games were aired on the state radio and recorded with hi-fi sets by hobbyists. Both scenes involved mail-order schemes. Later, when the only really global electronic communication system was operated by banks for wiring money, hackers left messages to each other in the depths of them. In the era of acoustic couplers dial-in Bulletin Board Systems (BBSs) provided custom interfaces for exchanging information in a wide array of formats. The formation of the Internet and its evolving set of community-developed protocols (from TCP/IP to RSS) can be seen in retrospect as the formalisation, unification and crystallisation of these efforts (in parallel with the increasing hegemony of the IBM PC compatible hardware platform). Beyond a certain threshold — as the user bases began to grow and the technology began to mature — base research slowed down and innovation was transposed to the higher layers of abstraction.

Once this happened, the Internet became the primary means of communication and data exchange between free software developers. It allowed people interested in very specific things to find each other and cooperate effectively across geographical barriers. Benkler formulates this in the language of economics as a drop in transaction costs, where cost ultimately stands for time and effort. In order to make most of the software/hardware problem later on, here it is crucial to establish a distinction between uses of the Internet for different purposes, if only on the analytical level of ideal types. On the one hand we can speak of exchanges in human language, which are common between hardware and software developers, and on the other hand there are exchanges in machine language, involving the sharing of source code and binaries, where the distinction becomes more interesting. Programming languages obviously share a lot of properties with human language — the most important being that they are immaterial forms of expressions —, so the infrastructure developed for human communication can be effectively used for the sharing of documents written in programming languages (with some modifications of course). The nuances involved with hardware design and development will be treated later on during the presentation of research results.

With the advent of the Internet it became possible to organise free software development using newsgroups or *mailing lists* (for humans) and FTP servers or software repositories (for computer programs). Thanks to a number of tools which integrated into this way of working, patches — modifications of source code — can be shared effectively from the distance from the convenience of one's workstation. A patch is a file which lists changes to the source line by line, and it can be applied automatically to the full source code. For example when a bug is found in a software, the person who provides the bug report sometimes attaches a patch which corrects the malfunction. Linus Torvalds, the project leader of the Linux kernel, is widely credited as having successfully organised in this way the largest free software project centred around a single program. Contrary to many reports, as Gabriella Coleman observed (2012), Hackathons and similar bodily gatherings still remained key to the functioning of most (but not all) free software development communities.

In fact this way of working did not change significantly since the 1990s, except for the adoption of *revision control systems*. Although revision control systems are as old as me, they gradually became more sophisticated, useful and widespread. Walter F. Tichy wrote the first one in 1982, called Revision Control System (knows as RVS), which is now part of the GNU project. This was little more than a set of scripts to keep track of changes in source code files. Its more advanced version was the Concurrent Versions System (CVS) by Dick Grune 1990 (or 1986), with project-level branching support. With branching, some developers could work on implementing a new feature while other developers made changes to the core functionality in another branch. When the new feature is ready, the two branches can be "merged". During the merge, changes are automatically applied to the relevant lines in the source, and human interaction is only necessary in order to "resolve conflicts" which arise when the two teams made changes in identical places. Such way of working increased the speed and flexibility of software development.

The next major change was introduced by the proprietary BitKeeper software, developed by BitMover Inc., whose CEO Larry McVoy was also a Linux hacker. Released in 2000, it was the first distributed version control system, abolishing the need for a central production server and allowing developers to keep and maintain a local copy of the source tree on their own computers or systems. This offered too many advantages to be reviewed here, but in general it is safe to state that it allowed more granularity (a.), modularity (b.) and loose coupling (c.) to be introduced to the free software development process. These features came as a life-saver for the Linux developers who struggled to "scale up" their operation to match the success of the project, the increasing number of developers and the complexity of the code that arose. BitKeeper granted a community licence (the BitMover Licence) to free software projects, which came with certain restrictions such as the ban on developing competing products and the ban on setting up independent BitKeeper source code servers. The adoption of Bitkeeper introduced a deep rift in the kernel developer community, with Linux Torvalds on one side who argued for BitKeeper on technical grounds, and Alan Cox on the other who argued against BitKeeper on political grounds. The boiling point was reached when in 2005 BitMover Inc. wanted to change the licence terms. As Steven Weber likes to say, characteristically the community found a technical solution to their social problem when they decided to develop their own alternative free software implementation. Torvalds developed Git, and Matt Mackall developed Mercurial. Finally, the kernel developers decided to adopt Git, and by now it is established as another legendary software which was published by Linus Torvalds, although the smaller foot stamp and the simplicity of Mercurial also found followers with important free software projects.

Git and the concept of distributed version control ushered in a new era of *social coding*. Developers could clone repositories more easily and start to make their own modifications, while at the same time allowing for merging the changes to the official version of the software if required. Communities could fork projects and unite them later if they wished. The versatility of Git lead to a series of new applications outside software development per se, which also affected the way free software projects worked. For example, people started to use it for cooperative writing, so software documentation could also be handled this way. Other people adopted Git as a back-end for storing versioned changes of Wiki pages, so that it became the storage technology behind many websites (for instance powered by the Ikiwiki software). Nowadays, git-annex extends the capabilities of Git to manage large binary files, and it is becoming an alternative to synchronisation and backup applications like the commercial Dropbox or Ubuntu's One service.

But probably the largest change have been the marriage of Git with Web 2.0 applications. The most popular such online platform is GitHub. Today, a critical mass of free software developers have an account on GitHub, which is a social networking site build around free software development, similar to the ancient Sourceforge. GitHub provides a backend for developers where they can push and store their changes in Git, and a frontend which provides a convenient web interface to the repositories. Features include statistics about the productivity of users and projects, automatic generation of online documentation from text files stored in Git, support for code review and discussions, contacting the developers and most importantly pull requests between developers. A common use case is that if somebody finds an interesting project which lack the features they need, they clone the project to their own account, make the necessary modifications, and send a pull request to the main developer. GitHub became a reference point on the job market as well, a way for developers not only for sharing their work but also for displaying in a universally understandable and comparable format their coding merits. Google and other companies are contacting and hiring developers proactively based on their GitHub accounts.

While these and similar tools have been heavily used by proprietary software developers as well, I argue that in the case of corporations and consortiums where there are more tight social structures in place to coordinate the work, they had a less significant impact on the software development process than in the case of free software development where people can (and do) make changes "all over the place". Furthermore, I argue that there is a structural homology in the technological tools used and the social process of production, or where such homology does not apply, controversies often arise. One example is the abovementioned BitKeeper fiasco.

The work of programmers is not necessarily directly useful for end users. Another type of worker has a pivotal role in the free software ecosystem: *the maintainer*. Free software operating systems come in *distributions*: a collection of programs that are tested and mended so they all work together harmoniously. Distributions have different *releases*, often according to a release cycle. The prototypical example is Debian, a community run Debian operating system (with a recently adopted release cycle of two years). However, in the case of Debian, at any one time there are three releases in use: *stable*, *testing* and *unstable*. Stable is typically used on servers which require absolute harmony, because it is the most well-tested variety. Testing is typically for end users who run it on their personal computers, and it could have minor glitches because not all the software is perfectly polished — the benefit is that software packages are more up-to-date so users can benefit from the latest technology. *Unstable* is for developers who are trying out the bleeding edge and who are not afraid is their computer is occasionally crippled by an update. They play an important part in developing the distribution because they often file and/or fix the bugs they encounter during their computing experience. These bugs are filed against the *packages*.

This is interesting to note because one of the widespread arguments about the advantages of free software is that it is more stable and contains less bugs than other software. While that is definitely true for the *stable* versions, we have to observe that the higher reliability is only possible because some people *want to run less reliable software*, to access new features or to find programming and system errors. In other words, free software is as stable as one wants it to be.

A *package* is a piece of program together with the meta-information that adapts the program to the specific distribution. It includes information like the name of the program, the name of the package maintainer who is responsible for it, the version of the program, and most importantly the dependencies: the names of other packages that are necessary for the correct functioning of the program. For example a word processing software may use an external spell-checker, so it would depend on the presence of that spell-checker program on the system. Packages can also include distribution specific changes to the source code itself (in the form of patches) and extra documentation. They also contain install scripts which set up the program according to the policies required by the specific distribution. Package maintainers take care of packaging these programs and uploading them to the repository. Packages appear in the unstable release first, and if they have proven to be working correctly in themselves and in conjunction with other parts of the system than they become part of testing, and finally stable. Figure 1. shows the whole process, only portion of which is explained here.

Apart from programmers and maintainers, the free software ecology includes a vast number of people who work in different capacities, for example system administrators, graphic artists, documentation writers, conference organisers, and so on. Figure 2. shows some of these functions inside the organisational model of the Debian distribution. The work of all these people is essential for maintaining and developing a working operating system. When we get to hardware, we will have to see if and how these or similar functions are taken up in the open hardware community.

In conclusion, it is not far-fetched to state that the means of production are in the hands of the free software workers. The necessary hardware and network connectivity comprises part of the default infrastructure in middle class households of more developed countries, while the software tools are themselves free software, legally available from the Internet. Additionally, there

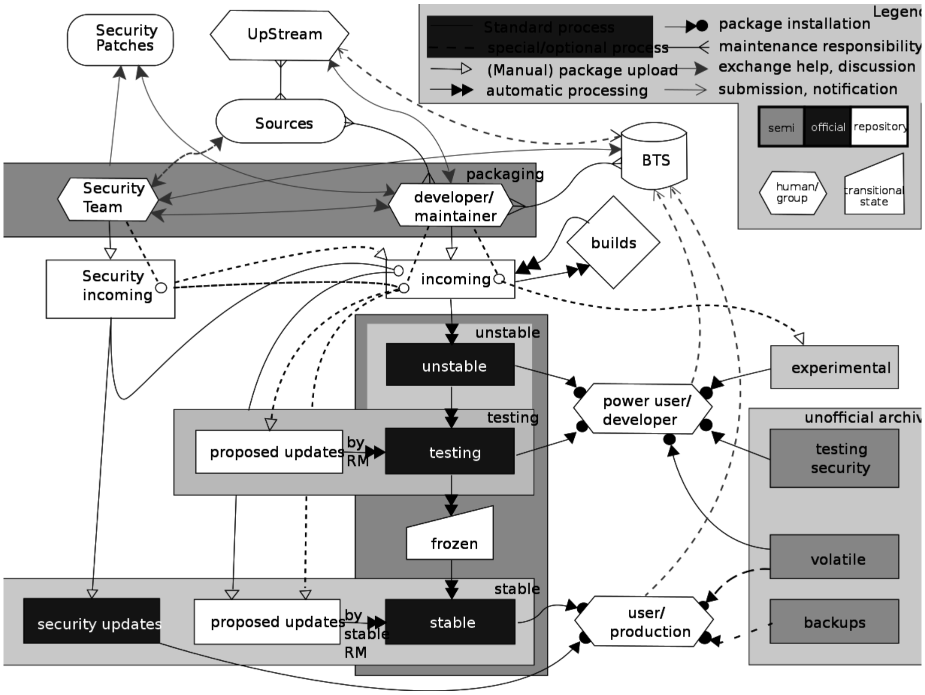


Figure 2: Debian package cycle. Author: Martin F. Krafft, based on the work by Kevin Mark. Licence: Creative Commons Attribution-Share Alike 2.5 Uported. Source: https://en.wikipedia.org/wiki/File:Debian-package-cycl.svg

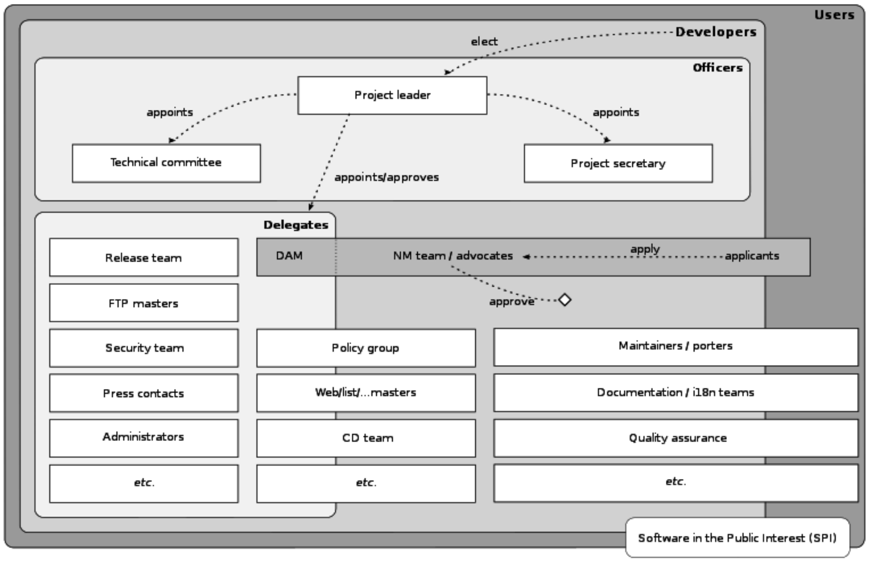


Figure 3: Debian organisation diagram. Author: Martin F. Krafft. Licence: Creative Commons Attribution-Share Alike 3.0 Germany. Source: https://en.wikipedia.org/wiki/File:Debian-organigram.svg

is a high level of social self-organisation of (mostly) workers, which structures and supports the production process. In the above description it is emphasised that the social organisation is deeply embedded in the technologies through which it takes place. The *solidarity* between workers is partly explained by the technological interdependence of the various development tools explained above: the programmers of web applications depend on the quality and suitability of toolchains, editors, operating systems – while in turn, package maintainers depend on the quality and suitability of the software that web application developers are churning out. Then, all of the above depends on the functioning of a free and open Internet.

In terms of social control and the maintenance of the status quo, such deep technological embeddedness is a two-edged sword. On the one hand, the traditional interpretation of technological power applies: the implemented technological measures define *de facto* what each participant can or cannot do, and they do so in a seamingly neutral manner. Since these measures appear in the life world as properties of the technology and the already configured infrastructure ("legacy" in computer industry trend) rather then as explicit results of political decisions, they don't appear in a political light in the first place, and hard to question in political terms. This is even more so because – as Söderberg sharply observes – hackers tend to formulate political arguments as arguments about engineering excellence. In late-modern society where science and technological rationality has great power to bear in rational discourse, this can be often successful. On the other hand, the hacker community (as well as the overlapping free software development community) has understandably a special sensitivity to power dynamics ingrained in technological solutions. While social power embedded in technologies can appear invisible, it is often exposed and can be read very precisely by people with the right background knowledge – the type of people who can actually build such systems. Söderberg phrases this as the question of what remains opaque and what becomes readable, and for whom, in the technology/power coupling. In concrete terms, are people whose agency is restricted by a certain configuration conscious of the historicity of that configuration or they see it reified as given as a finished fact of their life world? Since hackers and free software developers are the ones who make these things, and regularly get into political conflicts in the course of that making, they can more easily see and appreciate the political effects of technological decisions. What's more, – again in reference to Söderberg – they are often able to formulate their demands in the powerful language of technological rationality. Indeed, such a rhetorical move can perhaps be theorised to be the main impetus behind the comparative success of free software development and the strong ideological presence of hacker culture, relative to comparable social movements. Hackers are not only able to argue that what they demand is *right*, but they also argue, more persuasively, that it *works better*. Therefore, the increased technological embeddedness of power structures cuts both ways: while it does have a depoliticising effect, **it also makes power structures analysable as precise logical algorithms**.

Finally, there are two other aspects which received less attention in this section, although they are equally significant. They are symbolically speaking located above and below the life world of workers. Once, these are the legal frameworks of free software development: most importantly the institutions of the free software licences (Weber 2004, 1–20). Twice, hacker culture – including a strong ethical aspect (Himanen 2001) –, in which at least a vocal minority of free software developers participate. While the former has been addressed adequately by others, and the latter I will address bit by bit in the coming sections.

5.1.2 Lack of monetary compensation

While early scholarship on the free software community emphasised – or even celebrated – the free labour put into free software development by "volunteers", after some time social scientists had to adjust their analysis and admit that in fact there are masses of workers payed by corporations who sit day by day at their desks in boring offices, churning out line of code after line of code in the same way as their proprietary counterparts. **Understanding this contradiction led to a higher level of sophistication in the study of peer production, also teasing out the political differences between scholars.**

Both liberals and leftists had to explain the significance of the success of big companies like Red Hat, the large number of smaller web development startups and the army of freelancers who could make a living from free software. As Weber explains (2004, 195–207), there are many different business models built around free software, but the canonical example is selling services — mostly support, customisation and system integration — for free software applications. Such model is most successful when the entrepreneur manages to implement a mixed ecology around their core product. A mixed ecology means that the company takes the lead in development — investing resources which are seen by developers and users outside the company as a significant contribution, but at the same time it manages to build up a wide coalition of contributors outside the firm. This strategy depends on the balance between two advantages: one is that the company is seen as the single most competent actor on the market in relation to the given product, and the other is that the company receives the contributions of a vast number of distributed workers — for free. As both Harvey (2005) and Wallerstein (2004) argues elaborately, liberal capitalism only works well in a situation of relative monopoly. Under this business model, the monopoly position is not defended through the enforcement of an intellectual property regime or a similar systems of laws, statutes, grants, etc. The monopoly position rests on being the lead developer, and harnessing the reputation that comes with it (if done well). So the monopoly position is maintained not *de jure* but *de facto*.

The liberal interpretation from Benkler is that the rise to hegemony of such a regime necessitates the overhauling of the intellectual property regime, but does not endangers liberal democratic capitalism as a whole — indeed, it strengthens it. On the other hand, leftists like Bauwens (2005) and Rigi (2012) resort to the category of *articulation*. Articulation is the process through which one system of production can be productive embedded in another system of production — and possibly grow. The theoretical framework of their analysis is that a novel system of production is the basis of a novel economic, and therefore political system: cybernetic communism. Articulation allows cybernetic communism to express itself in the context of capitalism, like capitalism could articulate itself in feudalism before. It is an argument about "the new in the shell of the old". While Bauwens is more fatalistic or teleological, Rigi realises that whether the revolutionary potential of peer production is realised, and to what extend, is a question of the actual struggle, and not only theoretical explication.

Where these interpreters of peer production agree in their understanding of the "lack of monetary compensation" is a specific interpretation of what a *producer* is in the context of peer production. We adopt this understanding for the length of this thesis. On one end of the spectrum, producers are individuals who commit their "free time" or "leisure time" to a project, and their professional life is not connected to free software. On the other end of the spectrum, producers are wage slaves who sell their labour time in order to survive, and happened to be assigned tasks having to do with free software production. What brings them together is their productivity, which in turn requires a certain level of personal autonomy, easily — but not necessarily — translated into an experience of individual freedom and satisfaction. In contrast with the point of view of the individual explored so far, from the point of view of the organisation it is rather clear that a successful free software project have to build an image that is is worth contributing to, which usually rests on a mixture of technical rationale, cultural attraction, and sometimes political reasoning.

For instance, BackTrack Linux is a Debian-based live distribution "built by penetration testers for penetration testers". It is typically used a "live CD" or "live pendrive", which is not installed on the computer but boots directly from the external media. The *technical rationale* is that penetration testing requires both a clinically clean system (which is hard to achieve on a computer used daily) and a special set of software tools (which can interfere with everyday usage and take a lot of time to install and configure). Any live distribution (for example Debian Live) can provide a clean system, but not the special collection of penetration testing tools. Any computer can be set up as a penetration testing workstation, but then it is hard to use it for everyday tasks and it has to be "cleaned" after each testing mission. Therefore, a live CD for penetration testers fills a neat niche in the market of distributions. The *cultural attraction* of a community distribution is obvious. While Debian markets itself as the "Universal operating system", aiming for mass appeal, BackTrack Linux can cater for the taste of security specialists. The first target audience — conceived as the everyday computer user — will be scared, or at least alienated by black backgrounds and ninja iconography. However, BackTrack's slogans like "The quieter you become, the more you are able to hear." motivate many people to use or even develop it proudly, and provide a valued user experience for its target audience. Finally, BackTrack does not offer much in terms of political reasoning, other than stating that "it is free software and it will always be free software" — which ensures the community that their efforts will not be protected as intellectual property by the corporation in the future. In return of organising the BackTrack project, Offensive Security can position itself as a leading player in the security market and a competent partner of serious organisations who require its assistance.

The importance of these three factors can be demonstrated by cutting the user base of BackTrack into three ideal typical groups. *Professionals* use this distribution because of its aforementioned technical advantages, and what they get out of it is the results — they would often be workers in computer security companies which specialise in penetration testing. *Script kiddies* use BackTrack because of its aesthetic appeal which radiates adventure and competence at the same time. They benefit from the technical advantages it provides but in reality they could just install the relevant tools on their own computer — they are often amateurs and lack the skills to get the most out of what a specialised security distribution. What they get from it is the experience of "hacking", which build their identity in the eyes of their peers and their own. *Purists* could belong to either camp, and could be imagined as politically motivated hackers (or script kiddies) for whom using only free software is a political statement in itself. Even if they would not contribute much to BackTrack, they would potentially contribute to Debian, on which BackTrack is based. All three groups are essential for the success and sustainability of BackTrack (and Offensive Security) — even script kiddies who make the distribution known and spread half-truths about its notoriety, since the big clients on the market are often not more sophisticated in their judgement then the script kiddies. For a background image of BackTrack Linux, see Figure 3.

Figure 4: Example background image of BackTrack Linux. Author: BackTrack Linux. Licence: Copyright. Source: http://www.backtrack-linux.org/screenshots/

From the point of view of the individual and the life world, it has two consequences which are like the side of the same coin. On the one hand, it is possible to get involved in a technically sophisticated project without monetary investment, moving to another city, or an institutional background. This is a significant advantage compared to some other fields like molecular biology, for instance. On the other hand, it also means that without having a relevant job or professional carrier, the lack of monetary compensation is merely a form of free labour (Terranova 2000), and a hobby at best, where the life energy spent on it does not necessarily translate to a source of sustainable income. However, for serious developers – or at least aspiring programmers – it is a big advantage to be able to gain experience and skills in the context of a real world application, something which is often badly missing from graduate education.

From the point of view of corporations and capital accumulation, the situation is also somewhat precarious, as demonstrated above. However, the success of GNU/Linux and the companies which leverage it is a great example amongst many that opening contributions for the general public can potentially result in a product which is more competitive then anything the corporate players can cook up by themselves. Interestingly, the latest propaganda video (Foundation 2012) from the Linux Foundation (the current employer of Linus Torvalds) states that code comes from "about 8000 developers, from almost 800 companies", without mentioning to volunteer input explicitly. However, in their official white paper they are more nuanced, explaining the importance of unsponsored contributions in its complexity:

> There are a number of developers for whom we were unable to determine a corporate affiliation; those are grouped under "unknown" in the table below. With few exceptions, all of the people in this category have contributed ten or fewer changes to the kernel over the past three years, yet the large number of these developers causes their total contribution to be quite high.
>
> The category "none" represents developers who are known to be doing this work on their own, with no financial contribution happening from any company.
>
> The top 10 contributors, including the groups "unknown" and "none" make up over 60% of the total contributions to the kernel. It is worth noting that, even if one assumes that all of the "unknown" contributors were working on their own time, over 75% of all kernel development is demonstrably done by developers who are being paid for their work. (sic, "Linux Kernel Development: How Fast It Is Going, Who Is Doing It, What They Are Doing, and Who Is Sponsoring It" 2012, 9)

It is also worth to take note what the Foundation highlights in their presentation of Linux: the speed of development. They claim that no single company can throw so much development power behind a single project, which is quite right. This by and large supports the Shirky-Anderson thesis on the long tails [Anderson (2006b); Shirky (2003); BrynjolfssonHuSmith2003]. Anderson points out that given a discovery and distribution channel which scales well – made possible through Internet based platforms leveraging ICTs and automation – more profit comes from selling products in less demand than from selling products in high demand. The underlying – disputed (Elberse 2008) – claim is that related consumption patterns follow a power distribution, where the volume of the "head" is smaller than the volume of the "tail". The prime example is the online bookseller Amazon. The same seems to apply for production in large projects like the Linux kernel or Wikipedia. The ambiguity of these conclusions is that while the *number* of contributions is higher in the tail, at least in the production case the *volume* (number of changed lines) is lower in the tail and higher in the head.

In any case, there is no way to question that volunteer labour is a significant factor in free software development. The much bigger problem is that while on first sight it may seem that companies are growing a commons through their support, the ecology of business strategies is changing from support to services. *Google provides services using free software for its backend operations, but the resources and the frontend solutions are neither in the commons nor open sourced.* While Google have been criticised for not releasing its numerous in-house changes to GPL software like the Linux kernel, and indeed, it has changed its attitudes positively in the recent years, this is not the core of the problem. How much a company (or any other developer) contributes back to the community is a mere quantitative difference. The qualitative difference between Google and Red Hat is what is really important, and that's a difference between business models.

The core problem is what has been called the *ASP loophole* (Application Service Provider loophole). The terms of the GPL specify that a vendor can only distribute GPL software if the source code is included in the package, and changes to the software have to be licenced under the GPL too. However, Google and similar ASPs do *not* distribute software, merely use it to provide services. Therefore, they are not legally required to contribute their improvements back to the community – the ASP loophole makes them immune to the GPL. Of course publishing their inhouse technology improvements would provide an advantage to anybody who would like to replicate their service, such as their competitors, private users, or niche providers like radical technology collectives who support activists with trusted services. This is another way to develop a quasi-monopoly based on outsourced and uncompensated labour, but one that hurts the community rather than nourishes it.

Essentially, the momentum that the horizontal communication on the Internet as a communication commons meant on the networking layer, and the momentum which the free software movement has built up successfully on the software layer, is very much thwarted in the last few years on the service layer. Even if users increasingly run free software on their devices and connect to an Internet with net neutrality, at the end they use all that to reach the interface of a handful of corporate ASPs. While free software continues to receive increased attention, in actual reality the current political frontlines actually lie below and above the software layer: on the territory of services which are provided on the top of the software and in the field of hardware on which the software is running.

The abovementioned deficiencies of the GPL are addressed by the GNU Affero General Public Licence, or AGPL for short. AGPL version 3 was published in

2007 by the Free Software Foundation in conjunction with GPL version 3. There are three notable things to be mentioned about the history of AGPL. Once, it has been proposed by an ASP company (Affero, Inc.) with strong relationships to the free software development world. Twice, the first software published under this licence was *stet*, a collaborative online text editor which was used to write its own licence – one of the many references to recursion often found among hackers, who are obsessed with the idea of loops. Thrice, it has been used by some important projects (like the MongoDB database), but so far its adoption failed to take off. However, the promise of AGPL is to counter the updated logic of capital, and it is an important question whether open hardware licences make similar moves.

All in all, **the lack of monetary compensation is undoubtedly one of the most controversial features of free software development**. What is important about this paradox in a comparison with open hardware is that free software developers have found **a way to grow a common pool of resources while making a living**, or to put it another way, capitalists have found a way to exploit free labour effectively. In any case, it means that free software as a concept today denotes a sustainable and growing way of software production, in which, for better or worse, free labour plays a significant part.

5.1.3 Lack of chain-of-command

We have already seen in the case of the "Lack of monetary compensation" that the explication has to explore a more complicated reality than what is suggested by the title. In the first "naive" period it was almost inevitable that theorists would see the free software movement as an eminently horizontal political movement which is the polar opposite of corporations. A similar understanding reigned about the Internet: a network with a horizontal topology which makes it immune to state and corporate control, designed and run by information hippies who escaped the destiny of the military funding which backed them back then. A corollary of these misunderstandings was that free software projects emerge automatically out of the primordial soup of the Internet. As it happens, there is a core of truth in all these stereotypes, but for the sake of scientific understanding and political strategy it is vital to go beyond them and develop a more nuanced understanding.

The most important point is that — as Figure 2. illustrates above — it is is not uncommon to find hierarchical structures in free software projects. What *is* uncommon is to find chains of command, which essentially means bosses or generals who give orders. This goes against the core tenets of the free software culture. As the infamous IETF (Internet Engineering Task Force) slogan goes: *"We reject: kings, presidents and voting. / We believe in: rough consensus and* *running code."* (Clark 1992). Even if a profit-oriented company has ultimate control over a product, in case they want to engage with the free software community, they have to implement a governance structure which respects these principles. Managers inside the company can issue orders to their workers, but the community project itself cannot be governed so.

What keeps leaders from overreaching their prerogatives, corporations from overexploiting the commons, or development communities from disregarding the actual needs of their user base, is the practice of *forking.* Forking means to take the current state of the code base (the software as developed until a given point) and start a new project which takes it in a different direction. Especially since the advent of version control systems mentioned above, a fork is not necessarily a final or absolute decision. Projects can fork once and merge at a later point, or what is more common, developed in parallel while adopting patches from each other. The latter case means that some of the features developed in one project can easily find their way to the other without much duplication of code and programming effort. Forking is made possible by the free software licences which make the code base available for anybody to use. Therefore the fork is rather simple technically and legally – its main dread is that it divides the development community and the user base. This is sometimes called the problem of fragmentation. Since human labour and the market share cannot be duplicated like the code base, these are significant losses and they constitute a major danger for a free software project. The practical and political consequence is that decision makers (be that project leaders, corporations or development communities) are at the mercy of the people they are supposed to serve. This is a much more healthier political ecology than chains of command, since dependants have considerable power.

In an ideal typical free software project, participants take on responsibilities voluntarily, so they are called volunteers. In practice this even applies to some extend to wage workers, who often choose to work in the free software industry out of personal motivations. On the other end of the spectrum – which often falls out of the limelight of theoretical treatments – is an army of hobbyist, pet project developers, and learners who develop small programs for their own use and enjoyment. While some of these take off to become "organised" free software projects, by far the vast majority remains as a personal project, often clunky, unfinished, unoptimised or badly coded – but at least increasingly published on social coding sites like the abovementioned GitHub. These projects are not to be underestimated. They serve as educational resources for the people who work on the and also for people who discuss them personally or tumble on them when looking for a particular solution for a small problem. Developers learn, solve personal or particular business problems and prove themselves through these projects. Importantly, these pet projects are often ideal typical in a certain respect: people work on them according to their own life rhythm, with minimal interference from hierarchies, chains of command, user requirements or fellow developers. In this sense they provide an experience that developers would seek when they get involved in more sophisticated projects.

Evidence of this can be found as deep in the corporate heart as the project management literature. New trends in project management like agile and extreme project management emphasise the role of the manager as a person who is responsible for providing the best environment for programmers to do their job rather than focusing on the correct implementation of specifications and the timely adherence to schedules. Deliverables are shipped in tight incremental loops rather than all at once as with the waterfall model. Prototypes are tested regularly tested and evolve stage by stage towards the final product. Such an approach naturally involves a higher level of engagement and more commitment from the client, and puts more work on the manager in terms of dealing with human interactions. The claimed benefits are that programmers can focus on their work better, while the project as a whole can prevail in the face of shifting requirements and architectural uncertainties. Critiques point out, however, that such a development strategy is prone to fall prey to complex architectural problems that can only be tackled properly given the whole picture. Perhaps the emergent free software development model which features a random number of relatively uncoordinated contributors under the supervision of a strong project leader is an answer to such concerns.

As Raymond (1999) notes, the lack of a chain of command multiplies intelligence by giving more actors in the ecology of the project the power to make their own decisions. From a liberal point of view this is of course similar to the market as a coordinating mechanism, while from a leftist point of view an example of workplace autonomy and self-organisation. In any case, the crucial architectural consequence of the lack of chain of command is that intelligence is pushed from the core of the network to the periphery of the network, as decisions are made by actual workers and end-users. Such an organisational model bears striking similarity to the architecture of the Internet, namely the concept of the *stupid network*, or later called *dumb network*. First defined by Isenberg (1997), the basic idea behind such architecture is that information processing and decision making happens at the endpoint by the receiving and sending devices rather than on central processing units sitting in between. This allows user-end innovation without making costly and slow changes to the way the network functions, like upgrading the intermediary devices. Again, the argument can be made that the social structure and the technological structure are homologous.

5.1.4 Modularity

Modularity means that complex problems can be broken down into a number of simple tasks. In fact the theory and practice of doing just that is widely taught in programming classes, and generally deemed an essential programming skill. Of course not all complex problems can be attacked with this strategy. Although modularity can be usefully applied to complex software projects, it does not mean in itself that developers can simply drop in and out of larger efforts. As Brooke's Law (Brooks 1975) states, *"Adding manpower to a late software project makes it later."* The reason is that there are many solutions to the myriad of problems that a project presents, but it will only fit together if these solutions are compatible with each other. There is a strong cultural element in software development which can only be picked up through the immersion into the discourse of the specific programming community. Newcomers have to be integrated into the specific culture which developed around the project, otherwise they will cause more harm than they can help. On a larger level there is an idiosyncratic culture building up around each programming language, with its own set of cultural practices and orders of worth. There are many ways in which a specific problem can be solved in Python, but Guido von Rossum, the lead developer of Python, insists that "there should be only one [obvious] way of doing it". In a sense each programming language can be interpreted as a specific take on the general problem of programming, since on an abstract theoretical level all programming languages are by definition *Turing complete* — they can compute everything and anything that what is called a computer can compute. The difference is in *how* they are going to do it. In line with these differences, solutions which are not in tune with the culture of Python programmers are deemed *unpythonic*. The techniques described in this section have been invented to fight these cultural problems — mostly via technical means.

*Standards, file formats, communication protocols and coding style guides*are valued so much by programmers exactly because they work against such *interpretative — or more precisely design — flexibility*. These also help to implement *granularity*, which is discussed in the next section. They can be interpreted as the stabilised imprints of a certain thread of cultural practice. It is indicative of the gravity of the problem that in a certain sense the development of standards presents the same problem, only larger, as working without standards in a concrete medium-sized project. Differing opinions and cultural backgrounds have to be harmonised and a consensus achieved. Standards aim to be doing this for the specific area of the industry they target. They aim to lay the imprint of a general consensus, or at least something that can have the chance of being accepted in time as a general consensus. To build such a consensus and thereby to stabilise a technology is notoriously difficult and viewed as a great achievement and a serious contribution. Of course, standards only become standards worth their name once they are adopted by a critical mass of projects. Standards have some similarity with languages. Languages which nobody speaks are not very useful — in the same way that if everybody speaks their own language, nobody understands each other.

So why some programmers would spend time solving a bigger and harder problem — developing standards and other conventions mentioned above — rather then working to build consensus with the smaller number of immediate co-workers about their specific projects? The answer is *abstraction*. Abstraction is another basic programming skill, even more respected then the simple modularity presented above. Rather than solving a specific problem, abstraction develops a general solution first which then can easily be applied to solve the specific problem. The advantages are more or less obvious, and the disadvantages are illustrated in Figure 4. Even though developing the general solution often takes more time, it can then be adopted to similar instances of the same problem and more easily reused by other developers in different contexts. In fact abstraction goes two ways: it is the generalisation *of* the problem, but also generalisation *from* the context in which the problem occurs. The second aspect is occasionally called *portability.* Of course these techniques more or less apply to many areas of industrial design, engineering, or even knowledge production, but it is worth to note how they have been developed in the context of software development, which is in an inherently abstract area of engineering. The more interesting features of programing languages generally tend to be tools for abstraction, like functions, objects, macros and libraries. I will focus on the latter (*libraries*), since they have the biggest role in the free software development ecosystem.

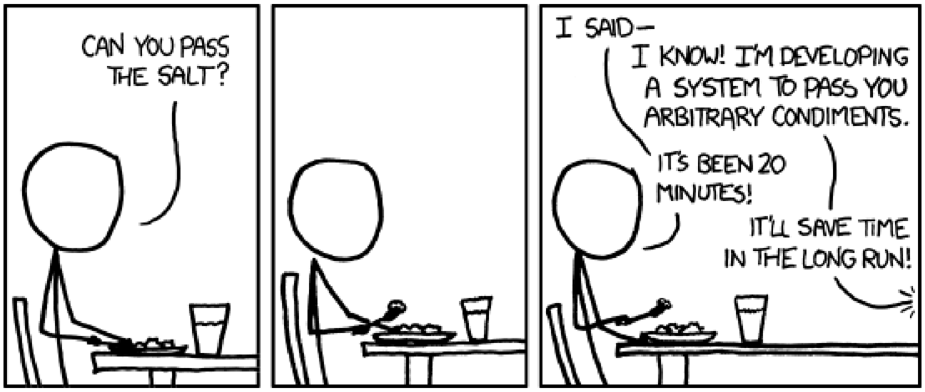

CAN YOU PASS THE SALT?
I SAID—
I KNOW! I'M DEVELOPING A SYSTEM TO PASS YOU ARBITRARY CONDIMENTS.
IT'S BEEN 20 MINUTES!
IT'LL SAVE TIME IN THE LONG RUN!

Figure 5: Abstraction: "The General Problem", a comic strip from the XKCD series by Randall Munroe. Licence: Creative Commons Attribution-NonCommercial 2.5 Unported. Source: https://xkcd.com/974/. Explanation: http://www.explainxkcd.com/wiki/index.php/974:_The_General_Problem

Without understanding the mechanics and politics of libraries, it is impossible to understand the idiosyncratic conditions of free software development. Since a library is a general solution for a specific set of problems under a given programming language, it can be isolated from the rest of the program and conveniently reused in other programs. For instance, in the Debian GNU/Linux operating system each library is installed only once, and the different programs which use the same library load it during runtime. This is called dynamic linking, which saves storage space and makes it more convenient to maintain the operating system. Dynamic versus static linking is the focal point of various sharp debates in the free software community around software engineering, information security, community management and licencing issues. Developing and sharing libraries — which usually happens in the context of developing a specific program — can be seen as a form of workers' solidarity between programmers. Separating the necessary functions from the actual program to a library and making them available separately is a great contribution to the free software ecology. Having well-maintained and powerful libraries available is key to the success of a software development project, because it reduces the required development time by fractions. In conclusion, libraries are one of the ways in which the abstract potential of software as an immaterial, infinitely copiable good is technically realised, resulting in code that is not only theoretically reusable, but can in practice be “built in” to a another free software project. **It is worth to note that modularity is therefore not merely an abstract or essential property of code as an immaterial good, but a potential that has to be realised through spending actual working hours first on architecture design and then on programming.**

5.1.5 Granularity

When hackers get into technology and find all the architectures potentially transformable through their labour, they often feel like the grandmasters of the universe. “For me, specialised and rarefied skills in technology is exactly akin to having super powers. You can do things that mere mortals are not supposed to be able to do.” This is widely reflected in the language of *philes* (Mollick 2005): text files which often describe the exploits of a group complete with the raw data which proves their credentials — usually the listing of contents on a supposedly secure server. Hackers brag and boast about their powers and ridicule their perceived opponents, often in l33t language – an obfuscated version of English where letters are substituted for printable ASCII (the most widespread character set) characters, and words are abbreviated to a single or a few characters according to numerous rule sets. An example is Gets Owned) (2012) a phile where the groups announces that they gained unauthorised access to the servers of a famous free software development group, Dyne:

```
f1rst up 0n th4 Xmas h1tl1st…th3 pr0ud kreat0rz 0f th3 dyne-bolic
distribution n s0me sh1tty ema1l cl1ent y0u hav3 n3ver herd of.
h0w sad. th3y ar3 push1ng s0ftwar3 4nd l1nux kern3lz wh3n d3y d0nt
3ven n0t1ce d4t th3y h4ve b33n backd00rd 4nd 0wn3d s1nc3 l4st
y34r! w3 h4v3 b33n enj0y1ng y3r c0d3 rep0z, dyne! th4nkz f0r th3
l4x s3cur3ty and 0utd4t3d k3rn3l!
```

However, as ensuing responses imply, l33t has lost its status from the language of the elite and it is seen by many as unnecessary one-upmanship. For instance Jaromil (a famous hacker and member of Dyne):

```
While the l33t sp33ch in the zine sounds quite l4m3 (c’mon guys,
its 2013, and happy new year!) the reader should be careful before
judging this as a scriptkid gig, because to our analysis it seems
to be an interesting hack.
```

Gabriella Coleman (2012, 106) makes the point that **as the hacker experience develops, hackers realise that the technology in which they are enmeshed is larger than any of them**. Similarly, Fuller (2003, 17) cites an exercise proposed by Donald Knuth (1989): to analyse every process that your computer executes in one second. All these authors reach the conclusion that no one individual can completely understand or control all the things that happen in a computer, so the individual desire for technological expansion runs up against a wall pretty quickly. **All hackers are at the mercy of other hackers** for support and help in tackling technological problems, and largely relying on systems which other hackers built. Therefore, with Coleman's words, *"Humility is as mandatory as arrogance."*

Building on that, it is possible to account for the fact that many advanced hackers are happy to help complete novices install GNU/Linux on their computers, as a gesture of initiation into the community. Problems encountered when fixing "consumer" Linux boxes for friends and acquaintances are treated as little gems if they have not been reported yet, and hackers often take pride in finding and reporting bugs in popular systems. Likewise, users are told and taught to report bugs properly. Unfortunately this is harder to do than many studies would suggest. On the next level, developers who use libraries and software frameworks in their work are expected to fix problems they find and contribute the code back to the community. Developers of those software rely on the user base as a source of valuable testing information. Since the general problems of technical architectures are large, there is a widespread understanding that everybody or anybody willing to contribute at least a small value should be made productive in the context of these efforts: hence granularity.

As explained in the introduction, granularity means that volunteers with a wide spectrum of motivation can make contributions to the project. This is made possible through a variety of technical means and social architectures. There is no need for special authentication for reporting bugs, and Debian even includes purpose-made bug reporting applications (like the reportbug package). Discussion of bugs mostly happens on public mailing list and actually the commitment for keeping them public is part of the Debian Social Contract. The goal of granularity is to maximise the number of contributors. The result is that the membership and boundaries of organisations are not clear, but more similar to the social movements where membership is largely determined by activity.

5.1.6 Loose coupling

Loose coupling is a principle of system design that is credited to come out from the ideas of McIlroy (1969). In conjunction with the larger idea, he also invented its practical germ: the *pipes*, which redirect the standard output of a program to the standard input of another program. It became a major type of interface between programs – the dynamic complement of text files, which are called the "universal interface", because they are easily parsed by programs and humans alike. As we will see in this section, the idea of loose coupling has to do with how larger parts of the systems cling together, in contrast with modularity which regulates abstractions between more closely tied parts like functions inside a single program or application. Economically, these techniques are essential for reducing complexity and reusing existing code, but their significance touches the basic tenets of system design. Through his work McIlroy also contributed to the development of what is called the Unix philosophy or the Unix culture, which is still a main source of inspiration for software engineers and system designers. He provided the most compact formulation of the Unix philosophy: "This is the Unix philosophy: Write programs that do one thing and do it well. Write programs to work together. Write programs to handle text streams, because that is a universal interface." (Raymond 2003, 34) Today many more sophisticated software frameworks are inspired by the loose coupling approach. For example the popular web development framework Django (written in the Python programming language) sports such loosely coupled components: the "persistency layer" is swappable, so the developer can choose to use text files or different kind of databased for storing information permanently. The benefit of the loose coupling is that the persistency layer can be replaced with a compatible one without having to change the other parts of the application, and if a new kind of persistency layer is developed, the integration costs are much lower than in the case of tightly coupled applications.

In terms of system design loose coupling is achieved through three basic steps. Firstly, the knowledge that one component of the system has to have about the system as a whole or the other components is reduced to a minimum. Secondly, a mechanism is implemented through which components can expose information to each other. Thirdly, an information exchange format is chosen in which the exposed information is represented. In recent years an increasing demand has arisen for microformats which standardise such information exchange. Open formats like JSON (JavaScript Object Notation, defined in RFC 4627, see Crockford (2006)) filled this space, enabling different web services and software parts to talk to each other easily. Any really mature and useful application being developed nowadays is expected to have an Application Programming Interface (API) which exposes its features. In the best cases this enables developers to use programmatically all the features of the application which normal users can use interactively, so that more sophisticated systems can be build on the top of it. Benkler and Raymond understands loose coupling as the reduction in *integration costs*, the time and effort involved in making different parts of a system to work with each other in an interchangeable and extensible way.

It is easy to see how these systems compare to the typical enterprise solutions developed by major corporations. Those are monolithic applications which try to solve all possible problems a customer might have within their own proprietary suite of applications. There is a running joke in the hackerspace of Budapest about the difference between enterprise computing and the Unix school of development: "Enterprise software is not ready when you have nothing to take away but when you have nothing to add." In case these closed systems have interfaces to other applications, they typically implement it using XML, the giant of microformats. XML is an extremely verbose and – at least by hacker standards – complicated way of representing structured data, and definitely more cumbersome to work with then the real microformats. In fact there is another joke about the Java programming language, which is popular in the corporate world but loathed by hackers for its slow and error prone operation: "Java is a program for converting XML to exceptions [that is, errors]." Another typical solution in the enterprise world is to invent obscure binary data exchange formats which are hard to debug, produce and consume. The reasons behind the inherent complexity of enterprise solutions and their extensive list of features – the latter of which is called "bloat" in hacker lingo – is not hard to understand. While hackers prefer elegance and flexibility because they want to build their own systems from loosely coupled building blocks, big industry players are selling programs which are marketed as a complete solution for a problem set, for example office computing. The more malevolent explanation is that if they sold a lean and efficient application, they could not get so much money for it than for an enormously complicated software suite with too many lines of code which needs a lucrative line of bug fixes and security updates. Keeping control of intellectual property is also easier if the systems are obfuscated and overcomplicated.

Loose coupling applies, mutatis mutandis, for social organisation in the free software ecology. The above mentioned bug reporting systems for instance make it possible for regular users who are not part of the software project per se to contribute bug reports and therefore take part in the development effort. *Bug tracking systems decouple membership in the organisation from filing bugs against the product.* Similarly, the institution of sponsoring packages in Debian decouples the right to upload packages to the official Debian repositories from contributing packages to the same repositories. Of course proprietary enterprise computing is, once again, more tightly coupled. For example many corporate platforms like Facebook require developers to register and get API keys tied to their accounts if they want to run applications on them, while most proprietary enterprise applications generally only accept patches, bug reports and other kinds of input from their own internal development theme. So while loose coupling is undoubtedly used in the enterprise world, it seems to be much more useful for free software developers and it is eagerly cultivated amongst their circles.

5.2 Participation

As mentioned above (in section C.), in the ideal typical free software project participants take on responsibilities voluntarily. Participants are said to be *self-selected*. In case somebody does not perform well enough in a specific role, community pressure is applied to the effect that another candidate should take over. In less important roles and in smaller projects, a role can be filled on a completely voluntary basis by any applicant, while in the case of more important roles and larger projects there are mechanisms of deliberation and the track record of candidates is taken into account. For instance, in various hacker gatherings the main conference programme is curated by the organisers, but there are always plenty of opportunity for participation and exposure outside the main track, often in the form of *lightning talks* where anybody can sign up on the day and have their 5 minutes of stage time. However, the most important difference between commonplace — and therefore commonsense — models of participation and free software development is that all structures are open to the general public and participants have a chance of actually performing the work which is associated with a certain role, and take on the associated responsibilities more formally once they have proved their fitness to themselves and to others. The logic (if not the reality) can be expressed in the aphorism "first do the work, then get the job".

This is obviously in contradiction with the common company practice of hiring somebody with the right skills to perform a specific job, although its advantages are also obvious. It is also becoming more popular in the corporate and academic world, in line with the latest development of cognitive or precarious capitalism. I will elaborate on the critique of productivity later on (in line with Terranova (2000)), but for now it is worth to note that under this new order productivity makes the subject: in commercial-academic terms, if you are not already producing you are not hireable, in terms of the hacker scene, it is your production which makes you a (potential) hacker. **To be a subject at all means to be productive.**

The way power and authority is distributed in these projects is often phrased by groups of developers or theorised by commentators as a form of *meritocracy*. As the idea that those with merit should hold key positions, meritocracy can be contrasted with plutocracy, where the rich lead, and democracy, where leadership is invested in the people. The merit of meritocracy is that it prevents the cumulative accumulation of power associated with the elitism of plutocracy, yet counters the populism which lets the unwashed masses near the reins. It is plain to see the contradictions of the libertarian political ideology here: hackers seem to be more liberal than the liberals, and go beyond the limitations of democracy, using a mix of left and right notions.

Analysed through the lenses of the left, meritocracy refers to a disguised idea of workers' self-management, the preference for managers to come from the rank-and-file and command authority through the technical prowess. Participation here is sanctioned based on one's contributions. Basically this is the criteria of a technical track record.

The intermediate — may be called technocratic or bureaucratic – criteria in meritocracy is qualifications, e.g. passing tests. This is rarely used in the hacker scene because of the mistrust of official credentials. However, increasingly complex organisations such as Debian project or the network of Independent Media Centers use formal criteria, application forms and examinations. Bureaucratisation and technocratism has obvious appeal to hackers, who deal with complicated formal systems and work with technology all the time. In a sense what any treatise on hacker culture has to explain is exactly who these forces cannot or have not yet taken over the scene.

Approached from the right, meritocracy grants authority to those who possess the right manners, just like in most elites. Of course the concrete contents of these manners are different. Usually covered under the rubric of the hacker ethics, these principles are still not adequately understood despite years of research. Himanen (2001) goes as far as suggesting that they are diametrically opposed to the Protestant Ethics of Weber (1958) which underpin capitalism. While Himanen's book was an absolutely necessary one, it is equally necessary to nuance his proposals and to reexamine them (as he does, too!). Especially because I see many common elements between the values promoted in the hacker scene and the old Protestant Ethic, and these common elements are more explicitly formulated and enthusiastically adhered to than in any other circles, including corporate cultures. One is exactly that the motivation for hard work – productivity – should not come from extrinsic factors but from a genuine interest. Such genuine interest marks the chosen ones, who are predetermined to go to heaven.

Putting these three together, there are two archaic models that come to mind (more or less from the right and the left, respectively). One is the *society of gentlemen*, signified by country houses and recorded for history in the letters, novels and biographies of the 19th century lower aristocracy. A gentleman does not have a profession – instead, he has professional interests, hobbies that he pursues at his leasure. Many branches of sciences found their beginnings in such endeavours, such as ornithology (biological systematics), numismatists (archeology), ethnography (anthropology). Moreover, the most important capital a gentleman possesses is good manners, which can potentially substitute any other, such as private income or social capital. Participation in the society of gentleman was tightly regulated according to rather vague rules, hinging on the perceived good standing of the individual. No wonder that in traditional English society persons ranking higher than the gentlemen were called peers, just like in the vocabulary of the BitTorrent file sharing protocol.

The other model which is more often compared to the meritocracy widespread in the free software development world is *medieval guilds*. This one also has a rich field of associations, since hacking is sometimes understood in terms of craftsmanship, and indeed, the heightened materiality of engineering formal systems (as compared to literature) coupled with the individual creativity required from the programmer makes a good comparison for crafts, especially in contrast to industrial production, and even design. Moreover, guilds have been self-organised associations of self-managed workers, where status revolved on demonstrated skills and competence, yet also a sense of duty and responsibility which enforced community norms beyond mere technicalities. While guild members sold their labour power and worked commercially, the guild itself also fulfilled a rich array of social functions from education through political rallying to social security. The masterpiece which allowed an apprentice to become a master craftsman (notably through mentorship) is comparable to the piece of software which earn a respected hacker her reputation. This is the aspect foregrounded by the notion that “hacker” is a term of appreciation and not an identity per se, and therefore it is not for one to decide if one is a hacker or not – the title has to be conferred by other.

All in all, meritocracy regulates participation and authority in basically two or three ways. Once, there is productivity; twice, following the rules; thrice, community ethics. One might ironically observe that this is not that different from any corporation: work results, administrative duties and adherence to the code of conduct of the corporation is all what is wanted.

In the **interplay of technical and social skills**, there is ample evidence which suggests that both are necessary. The most widely cited is the fork of the OpenBSD operating system from NetBSD, triggered in 1995 when Theo de Raadt was asked to resign from his role as senior developer and core team member in the latter project. The source of the conflict was de Raadts’ legendary square manners. While de Raadt is highly regarded as a security expert, his abuse of contributors and mishandling of conflicts earned him enemies. While forking is not necessarily a bad thing, it is worth to note that despite its technological superiority (at least in some specific areas), the BSD community have suffered from fragmentation through its history. It is possible to argue that Linus Torvalds’ ability to keep developers and stakeholders together around the rival Linux kernel – thanks to his widely acknowledged social skills – contributed much to its hegemony on the operating system market. In fact, Torvalds added a third, even more elusive criteria for successful participation in the free software community: *coding taste*. Coding taste refers to the programmers ability to make architectural decisions which are viable in the long term, write code which is understandable for others, and argue for his technical decisions in a clear, concise and convincing way. This last criteria unites the other two, since it is where human and technical problems meet: the field of aesthetics. The exploration of code taste is a prolific research area which cannot be expanded here, as it requires an attention to detail which is found more in literary criticism than in the social sciences, but it will hopefully be pursued in the context of nascent disciplines such as code studies.

The work of package maintainers was mentioned before in section A. According to the new policy adopted by Debian a few years ago, any member of the general public can become a package maintainer in Debian, provided that they find an *uploader* who already acquired the right to upload packages to the Debian repositories — which is a much more difficult process. The maintainer can choose any program which is compatible with the Free Software Guidelines issued by Debian and package them. It is the responsibility of the uploaders to check if the package conforms to the high quality standards which make Debian a respected distribution.

Exactly because Debian is respected for the quality — and great number! — of its packages, as well as the stability and security of the system, many people use it as a basis for building their own custom distribution on the top of Debian. The most famous of such is Ubuntu, built by Canonical Inc., a nonprofit founded by Michael Shuttleworth. Its aim is to make the system more user friendly and competitive against the mainstream, proprietary alternatives like OS X and Windows. However, there are many other distributions build with different goals in mind, many of which use Debian packages as their basic building blocks (like BackTrack mentioned above). Some of these are shown in Figure 4.

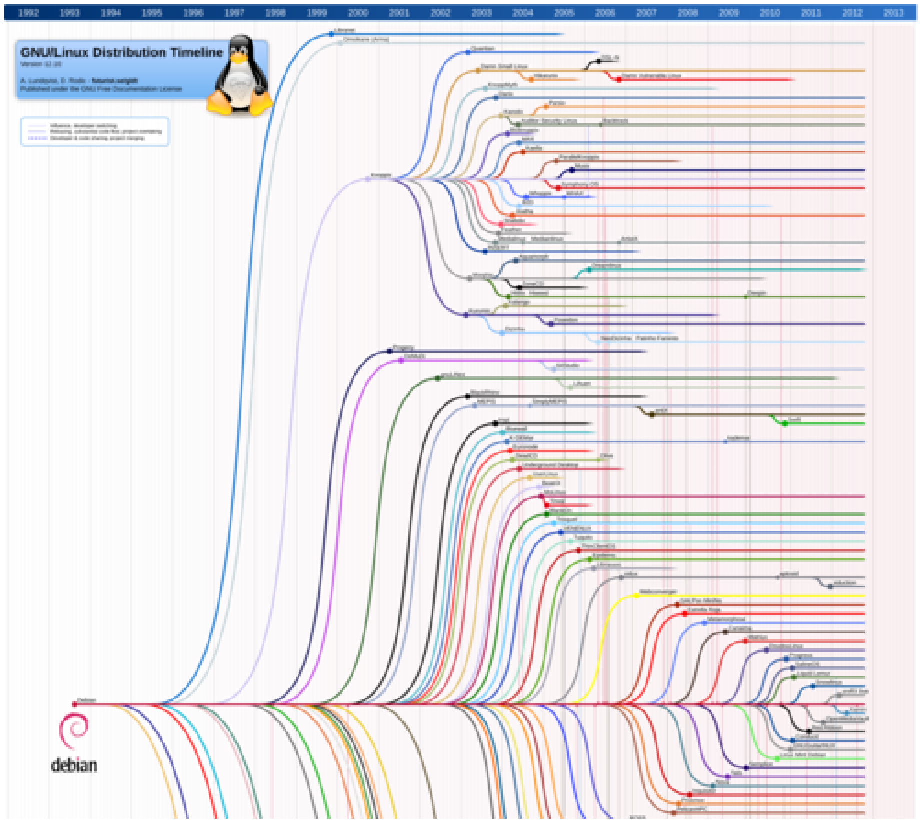


Figure 6: Some Debian distributions. Authorship: Andreas Lundqvist, Donjan Rodic. Modified by Michaeldsuarez. Licence: GNU Free Documentation License, Version 1.3 or later. Source: http://commons.wikimedia.org/wiki/File:Debian-FamilyTree1210.svg

Although the criteria of contribution may sound like the lowest possible barrier of entry, a section on participation would not be complete without considering the **barriers to participation**. Many virtual and physical places where free software is developed serve to some extent as a *third place* in the sense of Oldenburg (1989) – places to hang out and experience a sense of community apart from the home and the workplace. Where these virtual and physical spaces differ is exactly the concept of participation through contribution: where in other third places like cafés and barber shops participation happens through merely hanging out, conversing and sometimes consuming, production is the the most important factor here. On the other hand, as explained in the following section on expertise, knowledge of the field is not necessarily a barrier of participation. Furthermore, hackers don't need much capital to be able to contribute in some way, except that they need a lot of human time.

I write human time because one of the politically challenging and analytically difficult aspects of theorising hacking is that the time hackers spend in participating in development communities cannot be understood clearly through the dichotomy of free time or labour time, especially if the former is defined functionally as the reproduction of the latter. At least this explains why hackering can be inspiring and refreshing while at other times frustrating and exhausting.

In any case, having a lot of human time on one's hands for free software development can be good, bad or nothing special. Firstly, on the bright side some manage to earn money for doing what they also consider their hobby, or selling their talents for a high price so that they have to spend much less hours earning a living than most workers, or lead a fringe lifestyle which provides them with alternative means of subsistence. On the boring side, some are simply payed to work on a free software project which does not correspond very closely to their desires, or participate in a development effort as a way to improve their employability. On the bleak side, more and more are unemployed and/or disillusioned with the world of work entirely. In other words, diverse solutions exists to deal with the problem of time: autonomist solutions like squatting and scavenging, middle class solutions like adopting free software development as a hobby in one's *free time*, as well as capitalist solutions such as corporate sponsorship. This explains the diversity of actors in the free software world in particular and the peer production world in general, and the relative hegemony of these practices also accounts for their numerical distribution in free software development, for instance the predominance of professional, e.g. payed, labour amongst Linux kernel hackers.

Whether having time is an achievement, a privilege or a crisis, it is probably the most important factor in limiting participation. Many contradictions around the peer production model as it is used for the interpretation of free software development stem from the fact that the peer production model *requires* human time yet it *does not provide* a way of sustaining the developer either financially or through an alternative system.

On the bright side, provided that we take Feenberg (2002) seriously, and accept that **technology have to be qualitatively transformed through participation** if we want to escape the creepy technological rationality most sharply inscribed in the pessimist ontological determinism of Mumford (1967), Ellul (1964) or to a latter extent Heidegger (1993), the peer production model utilised in free software development appears to be a paradigmatic positive example. While neither Feenberg nor I agree with the ideas of the latter authors that (a.) technology has an essence and (b.) that this essence is a technological rationality which threatens life and the human development of civilisation, both of us are strongly convinced that the current technological paradigm is parasitic of life and ruinous for civilisation. Our main point of departure with the technological determinists is that technology, lacking essence, has enough flexibility to be transformed into a force which affirms life and which can potentially make a positive contribution to civilisation. Of course if that really happens is a matter that cannot be decided theoretically but only through empirical investigation and concrete political struggle.

One prerequisite is that the technological rationality which presents it as something neutral be challenged, and technology firmly planted as a central factor in everyday as well as institutional politics. The main line of attack at the moment is the development of research and social practices which revolve around participation. As Thorpe (2008; 2008, 76) notes, "[t]he shift in the orientation of STS and science policy studies is indicated by the primacy in contemporary discussions in these fields of the idea of *participation*." As I stated above, **hackers are a paradigmatic example of citizen participation in science and technology**, or – from another point of view – a significant edge case. While many STS authors discuss *consensus conferences* (where self-selected members of the public are involved in decisions of general interest about the deployment of certain technologies like power stations) and *citizen science* (where volunteers gather and analyse data in the context of a scientific research project), and some (like Latour) focus on *environmental protest movements*, these ideas and practices are inherently limited. These limitations are crucial not just in themselves but also because they limit both the scientific investigation of participation in technology and its enactment in the context of political struggles. In particular, I see three ways in which these lines of research and practice fall short of considering and realising the full potential of technological participation. For the sake of better words, **these practices can be criticised** as (1.) *coopted*, (2.) *managerial*, (3.) *reactive.*

Coopted refers to the idea of cooptation, when bottom-up ideas which are becoming popular are embraced by the establishment and turned into their opposite. I argue that when participation is organised from above and the framework of participation is designed by and integrated into the existing institutions, the full potential of participation can hardly be realised. This is because participation is limited to the content under discussion and cannot address the form of participation itself. In this reasoning I echo the line of argument advanced by Chantal Mouffe (for example in Mouffe (2000)), that the most significant struggles for democracy have targeted its very limits, e.g. who is included and excluded from the democratic debate and what form this debate would take. *Consensus conferences* are often organised by some arm of the state apparatus and generally conceived as producing input for the real decision makers – parliamentary politicians. On the other hand, *citizen science* is often organised by academic institutions and generally conceptualised as an effort by the citizenry to gather data which will be eventually evaluated and analysed by professional scientists. *Environmental protests* are generally not coopted, although sometimes it can be observed that they serve as a mere proxy between the institutions when they seek to amplify the impact of established scientific results, which they address to members of the parliament.

Managerial seeks to capture the attitude of controlling something without getting involved in its actual production. It is often a wholly legitimate behaviour since the stakeholders around a piece of technology are always wider than its producers – and indeed, it is most often the producers who are the less adversely affected. On the other hand, as Putt's law warns, "Technology is dominated by two types of people: those who understand what they do not manage and those who manage what they do not understand." (Putt 2006, 7) Many difficulties arise from the contradiction of these two concerns, but here I would like to point out a single one which is the most relevant here. *The managerial attitude is not well suited for the qualitative transformation of a technology.* Its input into the design process is often phrased in categorial terms of rejection or demand, while its decisions are not necessarily informed enough by the development practice, and therefore resonate poorly with practitioners. Consensus conferences which produce "outsider input" and environmental protests which reject whole technologies categorially (like nuclear power or genetic engineering) are prone to these weaknesses. Citizen science, on the other hand, often falls short by not being managerial enough: participants have little desire or possibility to influence research design, even though they are participating in it practically.

Reactive is my most serious concern about all three alternative practices which I contrast with the involvement of hackers in research and development. It is a commonplace that in terms of strategy, having the initiative is crucial. Yet, the alternative practices under consideration all too often react to an emerging trend or worse still, an established state of affairs. Of course struggles, innovations and inventions should start from the critique of the existing conditions, and no creative input manifests as a virgin in the midst of history. However, lines of attack bearing real impact should be transformative rather than reactive – that is, coupling their critique with creative alternatives. It is this kind of creativity – stemming from a freewheeling social milieu marked by competence – which is prevented by the coopted institutional framing of issues (in consensus conference and citizen science) and the reigning managerial attitudes (in environmental protests and consensus conferences).

To summarise, the participative potential of hackerspaces is that they elude the limitations of institutional cooptation, managerial attitude and reactive politics. **In hackerspaces people can develop, deploy, test and operate their own technologies free from these limitations.** At their best, these technologies become an active intervention in the existing state of affairs, proving the viability of alternative pathways of development or at least providing a proof of concept that currently widespread solutions are severely broken. These techno-political interventions can be an important component of struggles around technologies, especially if they are positioned strategically in the discourse. Of course, the main thrust of my research is to ask **whether such dynamics can work beyond free software, or how it is adapted to accommodate different technologies, and what new conflicts emerge from such an expansion to other fields.** The hackerspaces are the ideal site to study that issue because they mark the move of hackers from software to hardware, with recent forays to biology, etc.

All in all, while the free software development world in software, and hackerspaces in electronics, are currently one of the best sites to get involved in transformative technology research and development without institutional constraints, such radical **participation is constrained externally by the availability of human time, and internally by expertise**. The latter is the topic of the next section, exploring the social construction of expertise in the hacker scene (and especially in free software development) and how it is reimagined in conjunction with participation.

5.3 Expertise

Skills are the currency of the hacker scene. Yet, the hacker scene cannot be theorised as an expert community *per se*. In order to understand the role of expertise in it, a thorough reconceptualisation of expertise is needed. The main reason for this is that while in many expert communities expertise is primarily seen as *contributory expertise* – in the words of Collins and Evans (2007), in the hacker scene technological development goes hand in hand with technological education. Contributory expertise denotes the ability to contribute significant new results to a field of research endeavour. This is definitely recognised, valued and sought after in the hacker scene. However, **when assessing the status of an individual, the motivation and ability to learn, as well as the sensitivity to thinking out of the box, often takes precedence**. I tentatively term this *potential expertise*, which is a certain disposition for acquiring new knowledge. New knowledge here is referred to in a double sense: new knowledge can be new only for the individual herself, or it can be new for the community as a whole. This is in line with the emic understanding of novelty.

For instance there is a recurring conversation in the hackerspaces when somebody is reimplementing a small hardware hack that she saw on YouTube or the hardware hacking forums like Hackaday. When another person criticises the project as a mere copy, a customary answer is that this project may not be new in general, but it is new for the particular person who is undertaking it. Since difference is assumed between individuals and their environments, the next argument that comes up in such conversations that probably the eventual reimplementation will have some interesting derivations from the original design, broadening the state of knowledge about the subject. These differences are drawn from a wide spectrum, starting from the individual skills, taste and needs of the individual hacker, through the availability of materials and equipment in the actual hackerspace, ending in the correction of errors or actual technical improvements to the original design. Whichever kind is the derivation, however, it is still generally valued as a contribution, because there could be another user with the same skill set, style or needs; another hackerspace with the same infrastructural limitations; and of course actual correction and improvements always welcome. **So there is a constant oscillation between originality and reproducibility in the judgements about projects and artefacts.**

A similar example is my first encounter with a hacker who wanted to open a hackerspace in Budapest. He saw my Hungarian articles about hacklabs on Indymedia and contacted me, and eventually we became co-founders of the Hungarian Autonomous Center for Knowledge (H.A.C.K.). When we first met I asked him what programming languages or other technologies he is familiar with – I already had the understanding that skills were essential for setting up a hackerspace and a person's skill set is their main characteristic which determines their social position in the hacker scene. However, he basically dismissed the question as irrelevant. Instead, he preferred to let me know that **given a few weeks of intensive study he is able to acquire contributory expertise in any (or at least many) areas of information technology**. Thus, in this discourse – which I find informative about the general disposition of the scene – expertise has been constructed in a way that *potential expertise* – the ability to learn – is superior to *contributory expertise.*

The hackerspaces – which have been characterised as peer-to-peer learning environments by Raison (2010) – therefore mix research and development with autodidactic and community-based education, which fundamentally changes the social construction of expertise. As seen in this example, the line between research and education is often extremely thin and nondeterministic. A failed research attempt can still be valued as a learning experience, and what starts out as an exercise for learning can yield research results. Moreover, collaborations are often conducted with both goals in mind, with a more experienced person mentoring a novice. I claim that free software project are not fundamentally different in this regard. The reason why learning can easily lead to research is that learning trajectories are not uniform but highly individual and haphazard, in the style of autodidactic learning processes. It is easy to see that if not everybody learns the same things in the same way, but pursues their own track, learners can easily stray to previously undiscovered areas, even by mistake.

Another aspect of such dynamics is that people often encounter a personal problem whose solution proves useful for much more people, if the solution is general enough. See the discussion on abstraction in this same chapter on the idea of the general solution. Here it is worth to mention another example from the hackerspaces. The Metalab hackerspace in Vienna is pretty much open around the clock, sporting enough members in terms of numbers and diversity that there is always somebody hacking away at the premises. When these people become hungry, they often wants to order pizza or similar fixes. However, they found that most website in the area which offer food ordering services are broken in various ways, so they ended up writing and setting up their own service. It eventually became a successful spinoff company. Eric S. Raymond introduced the memorably phrase *scratching an itch* for such development dynamics as the first of 19 aphorisms in his book *The Cathedral and Bazaar*: "Every good work of software starts by scratching a developer's personal itch."

From the above phenomenon another characteristic kind of expertise can be distilled, which I even more tentatively call *differential expertise*. As we have seen, the status of such expertise stems from the widely realised fact that people and circumstances are different, yet this difference can be potentially exploited for universal benefit. The relative prominence of *differential expertise* in the social construction of expertise in the hacker scene is connected to the widely described fact that programming is perceived to be closely related with the arts and crafts on the one hand, and general aesthetic sensibility on the other. Therefore, as in the arts, individual contributors are not always treated as interchangeable, or even as comparable production units with individualised professional histories. This is a curious fact requiring much more attention, since in many other highly technical areas – take doctors, lawyers or physicists – the prominence of differential expertise is generally much lower.

The social construction of expertise in the hacking scene is characterised by two opposing cultural tendencies. One is best grasped in the discoursive formation of *eliteness*, while the other can be described as the ideology of *openness*. Historically, there is a clearly identifiable shift in emphasis from *eliteness* to *openness*. This is probably a general tendency which can be observed from punk through body modification to posthumanism: subcultures which grow into pop culture. However, both tendencies have their own specific expression in the history and culture of hackers.

Eliteness is associated with the modern myth of competence. The attraction of competence can be seen in how urban clothing is associated with (extreme) sports: for instance most people who wear Converse sneakers or Karrimor hiking boots rarely play basketball or go hiking in a serious way. Young boys are especially attracted to military aesthetics. Hackers take their clues from the competence associated with spies and the secret services, since many of their core technologies are used and have been developed in the context of signal intelligence.

While eliteness is historically receding, it can still be seen as an essential trait of the hacker sense, probably owing to the fact that it is centred around the growth and exercise of technical skills – so it should not come as a surprise that technical skills are, sort of, worshipped. Legends form around particular persons, communities and milieus that are kept alive in the written records and the casual discourse of the hacker scene. Stories of the old masters are retold and the news of novel exploits spread at hacker conferences and chat rooms. While bragging about one's own accomplishments is generally frowned upon – and, tellingly, often considered "old school" behaviour, discussing other hackers' achievements and contribution is one of the customary topics of conversation. The aesthetics of presentation in the hacker scene pull these two tendencies: the stress on skills and the surface of humbleness together. The key visual clue here is black. Black clothes – no matter if it is the anarchist black block style or the more elegant black suit as seen in the logo of the Anonymous movement – suggest outward simplicity and inward sophistication. Black backgrounds of websites and computer terminals suggest a depth ridden by mysteries hidden but to be deciphered by the chosen. The "hack center" of the largest European convention, the Chaos Communication Congress, is traditionally left unlit and relegated to the basement area. One can find people here who come to the Congress only to meet "specific people" and otherwise stick to their screens and keyboards unremittingly, disregarding the talks and presentations above. The saying "talk is cheap, show me the code" is characteristic of such attitude: work get done is valued above all.

Importantly for the social construction of expertise, in such a framework **skills are generally thought to be self-acquired, the autodidact a key icon in the hacker imagination**. The phrase *RTFM – Read The Fucking Manual* – is often quoted as the quintessential expression of such an attitude. It is used in the context of newbies ("n00bs") asking questions about something which they could just as well read up on. Another characteristic example is from the chat channel of the *suckless* (as in "software that sucks less") community, who write applications and libraries for people who write applications and libraries. They concentrate on minimalism in the implementation design and the number of the lines of code in the programs. Then there is someone called `bziur` coming to ask why there are no configuration files in their programs and why users have to edit the source code and recompile the program when they want to adjust a setting. The message from the regulars of the chatroom is clear:[11]

```
<morphles> this is not kind of comunity that lubes itself up to
get into someones ass
<morphles> Somehow people manage to get very stunted that they
just might not be target audiance of something
<bziur> No you lube to scratch your beloved egos. Hypocrites.
*** bziur (~bziur@84-10-217-19.dynamic.chello.pl) has left channel
#suckless: Leaving
<morphles> heh isint the one who thinks that all the shit should
be targeted at him is a with a kind of large ego
<morphles> oh
<morphles> he got away
<__20h__> I still don't get why config.h is somehow interferring
with my ego.
<morphles> :)
<c00kiemon5ter> "this is not kind of comunity that lubes itself up
to get into someones ass" hAHa :D
<morphles> c00kiemon5ter: glat you like it
<__20h__> morphles, yeah, that was a good one.
```

Note that `bziur`, who asked the initial question leaves the chat room in the middle of the excerpt: indeed, RTFM and associated attitudes are known to put users off, keeping the barriers of *elite* communities. This is, according to which side of the eliteness/openness debate are you on, a good or a bad thing.

Where eliteness and openness meet is the problematics that Collins and Evans (2007) points out in *Rethinking Expertise*. The main thrust of their work is to propose that in addition to the aforementioned contributory expertise there is another important type of expertise: *interactional expertise*. They define interactional expertise as the ability to pass as a contributory expert in a conversational about the field of the particular expertise. For example they test if the answers of Collins – a co-author and an STS scholar who studied the social aspects of scientific production in gravitational wave physics – to some of the usual questions of debate between gravitational wave physicists are distinguishable from the answers of an actual scientist working in the field. Here Collins – having spent time in the company of gravitational wave scientists – is assumed to have interactional expertise, while the actual scientists are assumed to have contributory expertise. As it turns out, Collins can pass as a gravitational wave physicist in conversation with other gravitational wave physicists, although he could not do the math required for most scientific contributions to the field. Interestingly, they model this experiment after the traditional Turing test, which is the oldest and still most widely accepted way to test for artificial intelligence.

[11] Chat log, irc://irc.oftc.net/suckless, 2013-01-22.

The Turing test aims to distinguish (and therefore, define) an artificial intelligence from a simple conversational program by asking ordinary humans to engage in a typewritten conversation with it. If the test subjects cannot distinguish between human and computer partners in the teletype conversation, the computer program on the other end is recognised to have artificial intelligence. Even though at the time of Turing (1950) teletypes were quite a curiosity, since the past decade IRC (Internet Relay Chat, a protocol for quasi real time text exchange designed in 1988) emerged as the main platform of social interaction in the hacker scene. Since the two communication systems share a lot of their essential interactional characteristics, hackers can be seen to perform never-ending Turing tests on each other. (Ironically, these chat channels also feature many automatic programs called bots which are indistinguishable for the first sight from human participants.)

As Collins and Evans point out, however, the Turing test does not differentiate between interactional and contributory expertise, and many people can acquire the former even in the absence of the latter by simply hanging around in the scene long enough. Therefore and especially in the context of the discoursive formation of *eliteness*, it is hard to see who has actual skills and who is merely talking like a pro. The *elite* response to this problem is that people are only allowed in the inner circles once they proved themselves in practice. This can be seen as a more or less commonsense *preselection mechanism.* However, the ideology of *openness* takes another route: people are allowed in to the lower tiers of production units more or less unconditionally, where they are given the opportunity to develop and eventually prove themselves *after* they are part of the community. This can be understood as a *post-selection mechanism*, where members are *filtered out* rather than *filtered in.* In this sense the contrast between these different strategies can be made in terms of *positive* and *negative* selection mechanisms, where the *elite* approach is associated with positive filters and the *open* approach is associated with *negative* filters.

Openness is a fundamentally liberal notion that found its way to hacker culture through the libertarian political atmosphere of the early hotbeds of the hacker scene such as the Artificial Intelligence Laboratory at the MIT and the amateurs of the Homebrew Computer Club. Levy (1984) makes an excellent job of describing these scenes – a book which became a classic of hacker bookshelves, enjoyed by generations of aspiring programmers. Fuelled by the since unprecedented Cold War spending on base research, at the MIT military funding and countless hours of overtime has been spent on haphazard projects, with graduate students having generous access to equipment and computer time. Meanwhile, in the Homebrew Computer Club hobbyists were labouring to “bring the computer to the masses" – a milieu that is often identified as an important point in the emergence of the personal computer. On the critical side, Barbrook and Cameron (1996) in their classic paper on *The Californian Ideology* describe how the social milieu and its practices were essentially exclusionary to women and minorities, while subservient to an emerging form of capitalism and supportive of the military-industrial complex, for example. While these shortcomings echo commonplace criticisms against liberal ideologies, Barbrook and Cameron (1996) also point out that libertarianism emerged as an explosive mix between radical left and right wing ideas — a notion too complex to elaborate here, although simply mentioning that *anarcho-capitalism* is an extreme form of libertarian politics may be sufficient to illustrate the fundamental contradiction involved.

The success of the *open* strategy relies on the fact that it can mobilise more resources more quickly than the *elite* approach. Of course this is not necessarily a requirement for every project. When a project or operation can be undertaken by a few dedicated individuals with the right skill set, openness is merely an overhead and the contribution of random individuals amounts to little else than noise. Another limitation of the *open* approach is of course the low amount of trust between the participants. For instance "release" teams who steal, digitise, package and distribute pirated movies usually work in small closed cliques, and the same can be said for most *black hat* hacker activities (harvesting credit card information, for instance).

In contrast with the elite approach, **open projects see education and outreach as integral part of their mission**, with the benefit that volunteers who receive essentially free education in these contexts will stick around for some time and enrich it with their contributions. Therefore, in the open projects skills are not treated as given, and while there is a great expectation for newcomers to educate themselves independently, the environment is much more supportive, mentorship more explicit and educational efforts more developed. For example, the new generations of hackerspaces have almost inevitably organise introductory courses on programming, soldering and the basics of electronics tinkering.

In fact one of the leading exponents of the hackerspace movement, Mitch Altman, is most respected as an educator. He spends much of his time travelling around the world and touring hackerspaces to hold workshops for all skill levels. The comic book introduction he collaborated on entitled "Soldering is Easy: Here's How to Do It" (Jeff Keyzer 2011) can be found in many hackerspaces – for instance the hackerspace in Budapest stocks a self-produced Hungarian edition (translated by dnet) printed and photocopied. In a similar vein, and building on these very efforts, during a hackerspaces workshop at the last Chaos Communication Congress it was evident from the introductory round that many hackerspaces are focusing on outreach towards teenagers, schools and families, bringing them in to the space and introducing them to the art of hacking. **It is clear that with the popularisation of hacker culture there is a shift of emphasis from the mere exercise of skills to their practical development.** As a result, expertise is not necessarily a criteria for membership in the communities, although it did not change that motivation and readiness is a key factor. Or expressed in a more nuanced way, it is fair to state that expertise functions as both the limit and the product of participation in the hacker scene, from free software development to the hackerspaces.

5.4 Closing remarks

In this chapter I tried to show that **through paying closer attention to the actual technology that hackers use to do free software development, it is possible to surpass the limits set by previous accounts and generate new insights**. I followed Benkler's schematic account of commons based peer production and attempted to flesh it out with ethnographic data and technical details. These observations are destined to serve as the basis for an explicit comparison when it comes to the problematics of open hardware in subsequent chapters.

In particular, I started with **a tour of the developer's toolbox**. This section showed how free software development is not merely conditioned by the Internet and by the immateriality of software code. **Software has to be built in a specific way in order to facilitate peer production.** The choices involved are choices about software architecture – a term that I propose to adopt from emic usage to my analytical framework. Moreover, the software code as such does not stand alone, but it is **immersed in a specific technical environment**, comprised of development methodologies, software tools such as version control systems, project management frameworks such as bug tracking systems, etc. Then, software is packaged for users in distributions of operating systems such as Debian GNU/Linux. The Internet and the immateriality of the software is concretised through these *architectures* into forces which form the real environment of free software development. While proprietary software development makes use of many tools described above, I argue that they are used in a different way there.

In the following sections I focused more explicitly on the **social architectures which condition free software development**, in particular on the social construction of *participation* and *expertise*. I argued that participation is a significant research interest from the point of view of a qualitative transformation of technology, and that the hack scene can serve as a paradigmatic example, or at least a significant edge case in this regard. I went on to analyse participation, which is based on *filtering in* or *filtering out* self-selected participants. These two mechanisms were called *eliteness* and *openness*, respectively. I noted the rising historical hegemony of openness within hacker culture, and its prevalence in free software development in particular. In terms of the limits of participation, I identified *human time* and *expertise* as the external and internal limits of participation.

The last section unpacked how expertise plays out in the hacker world. IT turned out that although expertise is central to hacking, the hacker community

cannot be understood as an expert community. One reason is that **learning and contribution to the field cannot be separated** – as in the case of the academia where students learn and researchers contribute, for example. This is why expertise can feature as both the limit and the product of participation. Analysing the various forms expertise can take, I identified *potential* and *differential expertise* as distinct factors in the social construction of expertise in the hacker scene, proposing them as new concepts for continued analysis.

6 Diachronic view: From Hacklabs to hackerspaces

Hackerspaces — and hacklabs, their predecessors – serve as the basic infrastructure for embodied communities engaged in peer production: “hackerspaces exemplify several aspects of peer production projects’ principles and governance mechanisms” (Kostakis, Niaros, and Giotitsas 2014). Therefore they are privileged sites for the study of peer production processes, since embodied communities can be studied using traditional ethnographic methods, and everyday practices can be reconstructed with relative ease. Moreover, they serve as the tangible infrastructure for building open hardware, the subject of this study (Maxigas 2014c). While much collaboration on open hardware happens online, sharing tangible production tools and tacit knowledge is only possible when hackers are together in their bodies. These two factors enable achieving the objective of looking at open hardware beyond licences as a social practice. Since FLOSS software development is as common in hackerspaces as hardware production, the differences between the two sets of practices can also be discerned.

In the narrative of peer production, hackerspaces are the closest to implementing peer production as a form of life – if peer production is an emerging mode of production, then hackerspaces are its factories, and perhaps its engineering schools and research centres. For instance, Moilanen (2013) discusses as the centrepiece of the *Emerging Commons Design Economy.* Showing the contradictions of technological fundamentalism (Chan 2014), fountainheads of the Californian ideology imagine hackerspaces to replace or complement libraries (Gershenfeld 2005), yet place them at the basis of a “new industrial revolution” (Anderson 2014). In this veritable liberal narrative, technologies developed collaboratively would eventually become household appliances in middle class nuclear family home. Meanwhile in South America, the FLOK Society charged with advising on the reorganisation of the Ecuadorian society on the principles of peer production recommends hackerspaces to augment the “commons-oriented productive capacities” of the nation, starting from building local communities. (Dafermos 2014) True to the principles set out in the previous chapters, this one takes the techno-futuristic claims of “fiction science” aside and looks at current realities from a critical point of view (Troxler and Maxigas 2014). Moreover, in order to deal with such inflated claims, it is useful to look at the prehistory of hackerspaces first, which served similar roles for the autonomous movement around the turn of the millennium, and can serve a historical counterpart for hackerspaces. While hacklabs were undoubtedly a form of life, their clear political orientation and their consistent critique of private property sets them apart from hackerspaces. I evaluate the advantages and disadvantages of both models and debate which one support peer production practices better.

In summary, this chapter introduces the research sites – which (as we shall see) are themselves *unfinished architectures* in their own right – and laying the groundwork for the case studies to come. Understanding governance structures, practices and imaginaries as well as (anti-)institutional histories is the mediating step between connecting the abstract theory of peer production with the case studies of small scale open hardware projects. As always, the analysis concentrates on the North European hackerspaces scene, which has served as a site of emergence, notwithstanding the fact that the hackerspaces idea has been appropriated with interesting divergences in the Americas and in Asia. In order to keep the analysis geographically and historically specific, hacklabs are surveyed as the ancestors of hackerspaces, and a short closing section outlines the subsequent waves of *shared machine shops* which followed the rise of hackerspaces.

One structural principle yet to introduce into the collection of architectural principles guiding the investigation is *shadows*. For each entity under consideration there shall be a shadow assigned which serves as a baseline for comparison. However, the principle is not comparison in the classic sense – the point is simply that at any one time it is only possible to understand something from a certain angle, and this angle should be set by a shadow. Unintuitively, such shadow comes first, e.g. it is a historical precedent. For open hardware – free software; for hackerspaces – hacklabs; for the r0ket case – mobile phones; for the door system case – time clocks; for the Arduino – enterprise servers. Such an approach locks out the possibility of simple "emergence", and forces the analysis to situate its subjects historically. In the case studies the shadows are commonplace electronic hardware, which allow engineering subculture to be related to mainstream engineering culture, bringing out its idiosyncrasies.

Therefore the discussion of hackerspaces begins with the story of hacklabs, proceeds with a more detailed look at hackerspaces as the research sites for the case studies directly following this chapter, and ends in situating hackerspaces in the proliferating genre of shared machine shops.

6.1 Hacklabs

Squatting as a social practice have engendered specific forms of live, producing a multiplicity of subcultures — hacklabs have been the site of an engineering subculture which developed in such milieus (Maxigas 2015a). Hacklabs were one of the first scenes where computer culture and political movements fused, forging embodied communities and fostering alternative practices of computing. While the history of hacking is crucial to understand these developments, here I focus on how this specific engineering culture fitted into its social, political, and physical environment.

The lucid definition of Yuill is a good starting point for understanding hacklabs:

> Hacklabs are, mostly, voluntary-run spaces providing free public access to computers and internet. They generally make use of reclaimed and recycled machines running GNU/Linux, and alongside providing computer access, most hacklabs run workshops in a range

> of topics from basic computer use and installing GNU/Linux software, to programming, electronics, and independent (or pirate) radio broadcast. The first hacklabs developed in Europe, often coming out of the traditions of squatted social centres and community media labs. In Italy they have been connected with the autonomist social centres, and in Spain, Germany, and the Netherlands with anarchist squatting movements. (Yuill 2008)

While each of these elements warrants attention, and indeed, shall be dissected further down the text, the main argument I am trying to build is that hacklabs fitted into a stream of autonomous politics and self-management practices that could develop outside of the modern institutional grid, proving that laboratories of engineering cultures are possible outside of the realms of state and capital. Hackerspaces grew out of the political frame but failed to integrate fully into the project-driven entrepreneurial culture at the bleeding edge of contemporary capitalism. Therefore my contrasting claim there will be that they are more driven by the semi-independent engineering culture associated with hacking. While in this section the convergence of hacking and radical politics is explored, the hackerspaces section focuses on the convergence of hacking and capitalism. The final understanding of peer production as a social practice will have to balance these tendencies and appreciate their particular trajectories.

6.1.1 History

The claim that hacklabs are a valid unit of analysis – e.g. that they hang together in reality enough to be studied empirically as a single thing – have to be itself substantiated. I argue that hacklabs have a consistent enough engineering culture and material practices because they share similar social circumstances and what I call a *scene*. A scene is made up of self-referential circuits of cultural communication and has vital online and offline components. It has its online and offline fora, its own jokes, language and history. It is not simply a common pool of knowledge but a common experience shared between people who mostly meet online but periodically gather in their bodies. Coleman already identified the "hacker con" as a central ritual of hackers, where solidarity is built, meaning is negotiated and efforts are directed in a common direction (2010). My observation is that people involved in hacklabs and hackerspaces are clearly invested in the general hacker scene too and they are overrepresented in hacker gatherings. Informants often report that the idea of founding a hacklab or hackerspace was born at a particularly inspiring moment of a hacker gathering.

Hacklabs existed basically since the advent of the personal computer, but their "golden age" has been the decade around the turn of the millennium. They have been most popular in Southern Europe (notably in Spain and Italy) and similar spaces in the North often had other names like "squatted internet work-spaces" or simply cybercafes.[12] However, given the remarkable consistency between the actual activities and their social context, I am discussing all of them under the hacklabs rubric. Nonetheless it is crucial to realise that there was a strong language and culture divide between North and South European movements and spaces, even if there have been various attempts to bridge it. The two most notable are the meetings of the Plug'n'Politix network[13] and the Transnational Hackmeeting in June 2004 hosted by the MonteParadiso a.k.a. Karlo Rojc squat in Pula, Croatia. Interviews with participants and organisers of these meetings bring out three key points: a. the strong motivation for bridging the Northern and Southern circuits of political hacking cultures; b. the perception of a strong divide based on culture and language; c. the relative failure of a. because of b. It is harder to put a finger on the actual differences, but consistency of practices in the South vs. specialisation in the North[14] as well as a greater interest in security in the North vs. media production in the South stand out as often-cited factors which can still be experienced today. However, it is also clear that the history of hacking in the United States and especially the engineering cultures around free software have been a common heritage equally appreciated on both sides and therefore serving as a medium of indirect communication.[15] Having said that, the three scenes have been in continuous contact with one another – for instance the visits of their more prominent personalities attests to that. Throughout the years both leading figures of the German and the United States scenes turned up at hackmeetings in Italy.[16] (Wikipedia Contributors 2015)

For the Southern hackers these meetings have been organised annually in Italy since 1998 (Florence) and in Spain since 2000 (Barcelona, CSOA les Naus),[17] constituting the heartbeat of the scene. Since accounts of their history are increasingly hard to find, and this geographical area and time period of hacker culture is little researched – notwithstanding crucial grassroots efforts like (Ferrer 2014) in Spain and (Autistici/Inventati 2012) in Italy – a schedule of hackmeetings is available in Tables 1 and 2. A consistent feeling from interviews is that the Italian hackmeeting seems to enjoy more prestige than the Iberian one, for instance practices and participants move more often from Italy to Spain than from Spain to Italy. While there are no hackmeetings in North Europe, very interesting hackmeeting traditions exist in Spanish speaking Latin America,[18]

[12] While offering and accepting donations for drinks, these spaces never charged for Internet connection or other services.

[13] 2001 October 5-7, Zurich: PNP1 Connect Congress, hosted by Egocity squat; 2004 December 3-5, Barcelona: PNP2 Connect Congress, hosted by Cyberforat squat.

[14] Such that a squatter-hacker should also be vegetarian, for instance.

[15] I am indebted to darkveggy, groente, Patrice Riemens and others for these insights.

[16] Wau Holland in 1999, founder of the largest hacker organisation Chaos Computer Club based in Germany; Richard Stallman in 2002 and 2011, creator of free software and the "last of the true hackers" (Levy 1984) from the MIT; Emmanuel Goldstein in 2007, founding editor of the legendary United States hacker magazine 2600; Andy Müller-Maguhn, spokesman and board member of the Chaos Computer Club also in 2007.

[17] CSOA is short for "Centro Social Okupado Autogestionado": Self-Organised Occupied Social Centre.

[18] Notably in Bolivia, Mexico and Chile.

even though they are out of scope of the present investigation. North European hackers have a different circuit with other hacker gatherings, similarly bipolar like the Southern European one. There the German node is somewhat more prestigious than the Dutch. Since the Northern European circuit is more connected with the hackerspaces, it is dealt with later.

Table 3: Italian Hackmeetings

Date	Location	Venue
1998-06-{5,6,7}	Florence	Centro Popolare Autogestito CPA FI-sud
1999-06-{18,19,20}	Milan	CSOA Deposito Bulk
2000-06-{22,23,24}	Rome	CSOA Forte Prenestino
2001-06-{21,22,23}	Catania	CSA Auro
2002-06-{21,22,23}	Bologna	Teatro Polivalente Occupato
2003-06-{20,21,22}	Torino	El Barrio
2004-04-{2,3,4}	Genova	Laboratorio Buridda
2005-06-{17,18,19}	Naples	CSOA Terra Terra
2006-09-{1,2,3}	Parma	Buffolara 8 [occupied for the event]
2007-09-{28,29,30}	Pisa	Centro Sociale Rebeldia
2008-09-{26,27,28}	Palermo	Centro Sociale Occupato Ask191
2009-06-{19,20,21}	Rho	Centro Sociale SOS Fornace
2010-06-{2,3,4}	Rome	Centro Sociale Autogestito La Torre
2011-06-{24,25,26}	Florence	Centro Sociale Autogestito nEXt Emerson
2012-{06,07}-{29,30,01}	L'Aquila	Asilo Occupato
2013-06-{7,8,9}	Cosenza	area ex-officine FdC Cosenza
2014-06-{27,28,29}	Bologna	spazio pubblico autogestito xm24

Table 4: Iberian Hackmeetings

Date	Location	Venue
2000-10-{20,21,22}	Barcelona	CSOA les Naus
2001-09-{21,22,23}	Leioa	Gaztetxe de Udondo, Bilbao
2002-10-{04,05,06}	Madrid	Labo03, Lavapiés
2003-10-{24,25,36}	Iruña	
2004-{10,11}-{29,30,31}	Sevilla	La Casa de la Paz
2005-10-{21,22,23}	Menorca	Es Mercadal
2006-10-{13,14,15}	Mataró	CSOA La Fibra
2007-10-{12,13,14}	Gernika	CSOA Astra Gernikeko Gaztetxea
2008-10-{17,18,10}	Málaga	La Casa Invisible
2009-10-{09,20,11,23}	Madrid	Patio Maravillas
2010-10-{21,22,23,24}	Zaragoza	Cárcel de Torrero
2011-10-{21,22,23}	Corunha	CSOA Palavea

Date	Location	Venue
2012	Calafou	Calafou
2013	Pamplona	Euskal Jai / Iruñako Gaztetxea
2014	Marinaleda	

According to legend, the first hacklab was founded in 1995 in Catania (Sicily) of all places. Freaknet or Poetry hacklab continued to be an inspiration for hackers throughout decades[19] but what is more interesting for us is the point where hacklabs became a genre of initiatives recognisable in the scene. Oral history and the few written records we have pinpointed this moment to the concluding discussions of the 1999 hackmeeting in Milan (Italy). (ana 2004; Anarchopedia contributors 2006; anonymous 2010) After this hackmeeting many Italian and Spanish participants went home with the common understanding that they had to found a hacklab in their home town. Indeed, empirical data presented in Figure 1. which I gathered from hacklabs.org (a now defunct catalogue of hacklabs) based on domain registration years shows a steady rise in the number of hacklabs. The same graph shows an accelerating trend in the founding of hacklabs from 2003 which does not have a corresponding event in my collection of oral history, other than the largest hackmeeting ever in the Iberian peninsula with almost a thousand participants (Madrid, Labo03). The demise of hacklabs caused by changes in the social, political and technical context is narrated at the end of this section – here let it suffice to say that few were founded after the year 2010. Based on anecdotal evidence and desktop research, it is safe to assume that the overwhelming majority of them closed down by now, but it is hard to get reliable metrics to probe.

6.1.2 Activities

Often located in squatted spaces and occupied social centres, hacklabs were part and parcel of the autonomous politics toolbox, on par with such institutions as Food Not Bombs vegan kitchens, anarchist infoshops and libraries, free shops and punk concert halls (Maxigas 2012a). For instance, the *Les Tanneries* occupied social centre in Dijon (see Figure 2) housed all these activities under one roof at some point, as did the *RampART* in London, the *Rimaia* in Barcelona, or *Forte Prenestino* in Rome. The largest network of hacklabs existed in Italy,[20] where influential hacklabs bloomed from the LOA hacklab in the populous North (Milan), through Forte Prenestino and bugslab, also in Rome, to the already mentioned Freaknet. Today, notable examples exist in Amsterdam (LAG[21]) and near Barcelona (Hackafou)[22]. Both operate in the context of a larger autonomous

[19] Cf. the 2001 hackmeeting and the Museum of Working Computers in Catania now.

[20] Link collection from Austistici/Inventati: http://www.autistici.org/hacklab/

[21] http://laglab.org/

[22] https://calafou.org/en/proyectos/hackafou

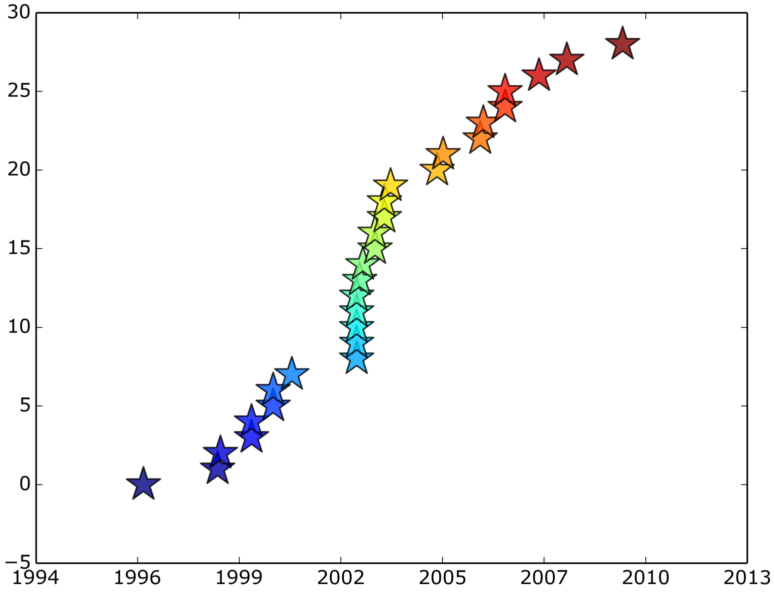


Figure 7: Registration dates of hacklab domains from hacklabs.org, based on the whois database. Own work.

space: the Binnenpret[23] in Amsterdam is a legalised (ex-squat) building complex which houses an anarchist library, the OCCI self-managed musical venue, a vegan restaurant and the Revolutions Per Minute record label, amongst other things like apartments; while Calafou[24] is an eco-industrial, post-capitalist colony based on a cooperativist model, including a social centre with a concert room, freeshop, kitchen, library and many other "productive projects". It is telling that neither host space is an illegal occupation like most houses which hosted hacklabs in their heyday. Since hacklabs themselves were spatially embedded in occupied social centres, and most of their participants lived in squatted houses, hacklabs were also socially embedded in this milieu. Hacklab participants routinely participated in other activities organised on site or in the city, such as solidarity concerts, recycling food from markets and dumpsters (e.g. "skipping"), occupations and other direct action, etc.

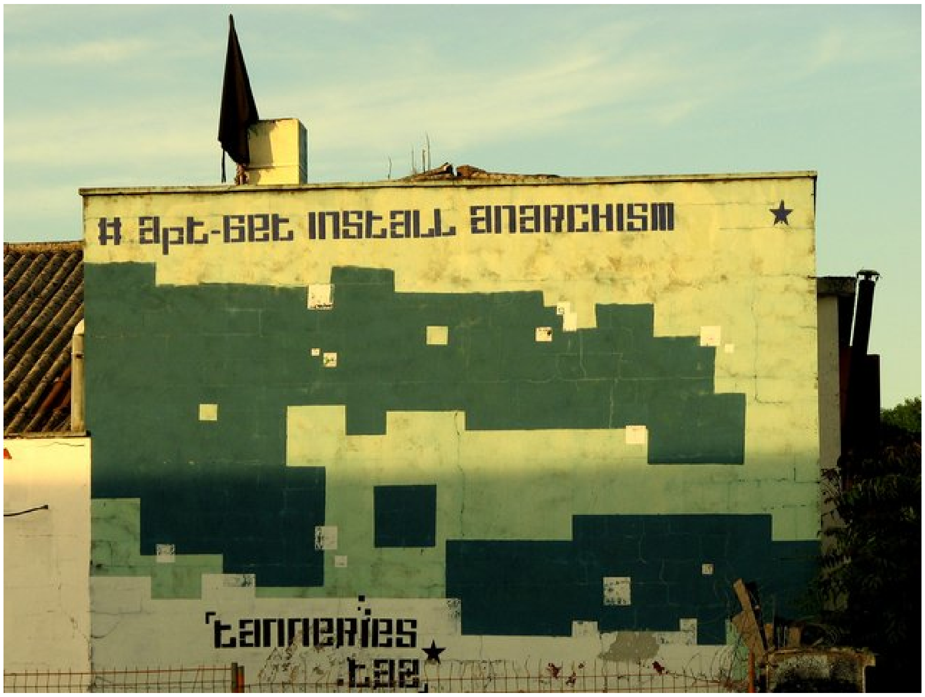


Figure 8: Les Tanneries squatted social centre, Dijon, 2007. Photo published by nigra. Source: https://linksunten.indymedia.org/de/node/98266 Licence: Creative Commons 2.0 Attribution Non-commercial Share alike Unported.

Since squatters largely work from recycling trash, in a way it is inevitable that when computers and networking equipment turn up in junk piles, they will be utilised in squatted social centres by grassroots activists. As a general rule

[23] http://binnenpr.home.xs4all.nl/
[24] http://calafou.org/

one can say that any category of goods which can be recycled from refuse will be put to creative use in squatted social centres. In the beginning of the 1990s computers became household electronics and in the middle of the decade modular IBM-PC compatible computers were not just ubiquitous in richer middle class homes, but enjoying a quick turnaround driven by incremental hardware upgrades. When personal computers were still unaccessible for the lower middle class, "[m]embers of the collective scavenged and rebuilt computers from trash" (Wikipedia contributors 2014a). Obsolete computers and discarded hardware would often find its way to hacklabs, and transformed into useful resources — or failing that, to artworks or political statements (Figure 3). Blicero from the LOA hacklab in Milan said that "We built a classroom of i486 PCs recovered from the dumpsters of banks and other offices." (Anarchopedia contributors 2006)

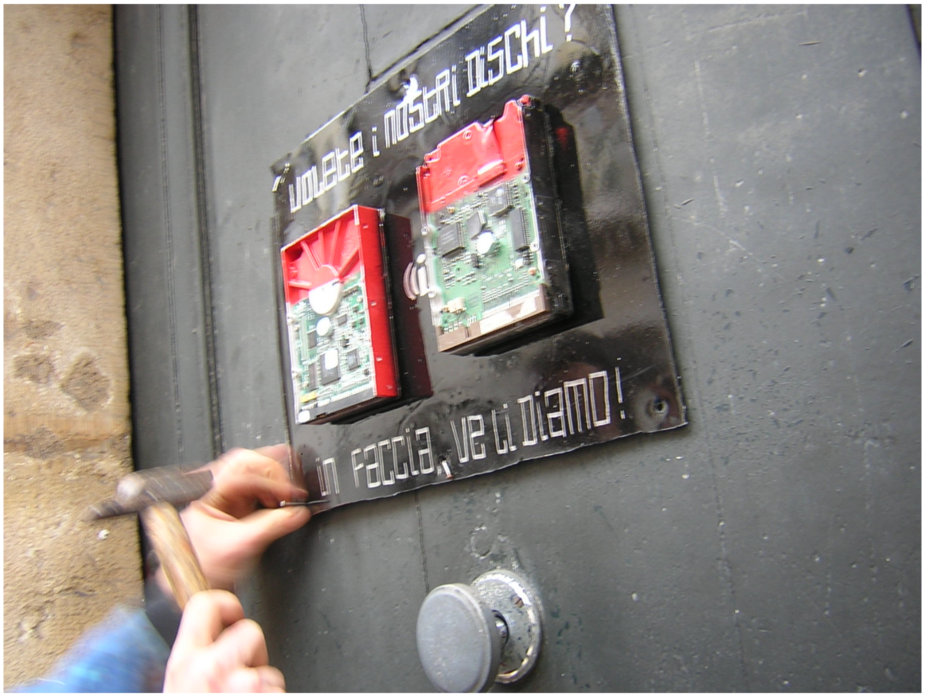


Figure 9: Old hard drives nailed to the front door of the police station in Dijon, France. Action against the censorship of the local Independent Media Center. 4 November, 2004. Photo published by print. Licence: Copyright. Source: http://print.squat.net/move.html

In the decade before GNU/Linux adoption achieved a critical mass, installing a FLOSS (Free, Libre, Open Source Software) operating system was an art or a craft, not a routine operation. In such a cursory moment, free software was not yet established as a lucrative segment of the market, but had some characteristics of a movement, and hacklabs housed many developers. Software support was a main line of activity in hacklabs, with squatters, activists and some members of the general public coming specifically to get help, and hacklabs like LOA were organising courses for beginners and intermediate users alike, while experts were collaborating in contributing to the software themselves.

While hardware came from the junk, and communities of practice formed around technical skills in occupied social centres, knowledge and software were shared over electronic communication networks. However, even access to these networks had to be established collaboratively and tied to specific locales. At a time when modem connections were considered modern, it was sometimes only possible to connect to the Internet (or its predecessors, like BBSs and networks like FidoNet) by getting down to a hacklab in your neighbourhood. Building and cracking wireless networks has been a key skill of hacklab participants, often requiring substantial work on the physical layer. Hacklabs became grassroots communication hubs. In the times before mobile phones and well before popular voice-over-IP solutions like Skype, hackers from WH2001 (Wau Holland 2001), Madrid and bugslab, Rome set up telephone booths on the street where immigrants could call home for free. Therefore, these "squatted Internet work-spaces" – as they were sometimes called in the North of Europe – did not only facilitate virtual connections between people and machines but also contributed to the formation of embodied counter-computing communities. *Ironically, Internet use brought people together in physical spaces.*

At the same time media activists seized the new opportunity brought about by cheap ICTs to produce propaganda and build alternative networks. Halleck (1998) emphasises that at least some activists started using ICTs as soon as they became available. However, access to knowledge was relatively scarce – especially outside the academic and corporate environments – so that autodidact users struggled to find associates. Marion Hamm observes that physical and virtual spaces enmeshed due to (Indymedia) activists' use of electronic communication media: "This practice is not a virtual reality as it was imagined in the eighties as a graphical simulation of reality. It takes place at the keyboard just as much as in the technicians' workshops [e.g. hacklabs – maxigas], on the streets and in the temporary media centres, in tents, in socio-cultural centres and squatted houses." (2003) In the early naughties the largest media activist network was Indymedia [Halleck (2003); Pickard2006a; Pickard2006b], and according to my ethnographic research most hacklabs were used by Indymedia activists at one time or another. Hacklabs provided the peace-time offline (and often online) infrastructures and the embodied communities which supported the Indymedia network and related activities.

One example of these four factors (1. junk, 2. FLOSS, 3. network access, and 4. media activism) coming together is the Ultralab in Forte Prenestino. Forte Prenestino is an occupied fortress in the heart of Rome which is also renowned for its autonomous politics in Italy. The Ultralab is declared to be an "emergent pattern" on its website (Avana.net contributors 2005), bringing together various technological needs of the communities supported by the Forte. The users of the social centre have a shared need for a local area computer network that connects the various spaces in the occupied fortress, for hosting server computers with the websites and mailing lists of the local groups, for installing and maintaining public access terminals, for having office space for the graphics and press teams, and finally for having a gathering space for the sharing of knowledge. Meeting these needs is not a light undertaking even by corporate enterprise standards, since the area covers 16500 square meters of shifting flotsam and projects run on a no-budget basis. The point of departure for the hacklab was the server room of AvANa, which started as a bulletin board system (BBS): a dial-in message board in 1994 (Bazichelli 2008, 80–81). As video activist Agnese Trocchi remembers,

> AvANa BBS was spreading the concept of Subversive Thelematic: right to anonymity, access for all and digital democracy. AvANa BBs was physically located in Forte Prenestino the older and bigger squatted space in Rome. So at the end of the 1990's I found myself working with technology and the imaginative space that it was opening in the young and angry minds of communities of squatters, activist and ravers (Willemsen 2006).

AvANa and Forte Prenestino connected to the European Counter Network[25], which linked several occupied social centres in Italy, providing secure communication channels and resilient electronic public presence to antifascist groups, student organisations, free radios, the Tute Bianche militant social movement, and other groups affiliated with the autonomous and squatting scenes. Housing servers inside squats had their own drawbacks, but also provided a certain level of physical and political protection from the authorities. While such setup worked out for decades, it is telling about the deterioration of hacklabs as an infrastructure for social movements that in 2012 a European Counter Network server was seized by the FBI not from an Italian occupied social centre but from a professional server farm in New York hosted by a social justice oriented Non Governmental Organisation (May First / People Link 2012). In fact autonomous server projects have been the few components of the scene which survived to this day, and as the ECN case shows they continued to operate services, but in a more professional way.[26]

The descriptions given above serve to indicate how hacklabs grew out of the needs and aspirations of squatters, media activists and to some extent other

[25] http://www.ecn.org/

[26] Still active collectives such as Autistici/Inventati from Italy, Poivron/Potager from France, Sindominio from Spain, or PUSCII from Utrecht started in now-defunct occupied spaces and now host their servers in more professional settings.

marginalised groups or even the general public. In broad terms these activities could be treated under the rubric of *access activism.* Access activism in hacklabs had a number of characteristics which are important to spell out clearly.

Firstly, that the hacklabs fitted organically into the anti-institutional ethos cultivated by people in the autonomous spaces. In the same way as free shops recycled clothes to serve as an alternative to commercial fashion shops, hacklabs recycled computers and taught and developed ICT knowledge as an alternative to computer shops, computer courses, corporate research and development. They did so without any official institutional support or backing, organised in an informal and horizontal way, along explicitly political aims and principles.

Secondly, they were embedded in the political regime of these spaces, and were subject to the same forms of frail political sovereignty that such projects develop. As occupied social centres typically have written and unwritten conducts of behaviour which users were expected to follow, in and out of the hacklab. These informal by-laws typically stated for instance that people who exhibit sexist, racist, or authoritative behaviour should expect to be challenged and, if necessary, forcefully excluded. Such rules created what was called the *activist ghetto*, where many mainstream attitudes were effectively outlawed, but at the same time the same rules created a “safer place” for groups with limited access to social spaces like (illegal) immigrants or queers.[27]

Thirdly, the political dynamics of squatting, and more specifically the ideology behind expropriative anarchism[28], had its own particular consequences. A social centre is designated to be a public institution whose legitimacy rests on serving its audience and neighbourhood, if possible better than the local authorities do, by which the risk of eviction is somewhat reduced. Thus the open door policy of hacklabs and the low barrier of access in terms of credentials or skills is mandated.

Lastly, the state of occupation fosters a milieu of complicity. Consequently, certain forms of illegality are seen as at least necessary, or sometimes even as desirable. These factors are crucial for understanding the differences between hacklabs and other *shared machine shops* like hackerspaces. For example in the latter case illegality is much less embedded in the social context of the space (because it is rented and operated by a foundation), allowing for certain projects (like spin-off companies) which would be impossible in hacklabs, and making some normal hacklab practices (such as stealing wireless Internet from the neighbours) regarded as suspicious.

[27] “Safer places policies” have been used in London social centres.

[28] The idea that one can take (e.g. appropriate) the resources necessary to realise higher social aims even if such acts are deemed illegal by the establishment. In practice this usually means “finding a better use” for unexploited resources, but sometimes it could mean stealing them from powerful owners and putting them to a more social use too.

Hacklabs seamlessly combined three functions: providing a social- and work space for (underground) technology enthusiasts to learn and experiment; supporting and participating in social movements; providing open access to information and communication technologies for the public. In cyberspace, everything was still fluid and there was an overwhelming intuition, paradoxically inspired by cyberpunk literature, that if the losers of history learn fast enough, they can outflank "the system". Paradoxically, since all major cyberpunk stories described a dystopia where corporate power incorporated state power and runaway technology has become the scourge of civilisation, without any hints at a real change through either technology development or social movements. Such techno-optimism was not altogether unfounded, however. It is important to remember that before the dot com boom[29] neither state nor capital paid serious attention to the Internet, yet it seemed to offer unbounded possibilities to any young person familiar with sci-fi. While the autonomous movement in general was waning away, cyberpunk lived its golden age.

In conclusion, hacklabs were political projects that appropriated technology as part of the larger scheme of the autonomous (squatter) movement to transform and self-organise all parts of life.

6.1.3 Demise

Access activism as it was became largely obsolete when Internet connections and basic networking equipment like routers and IBM-PC compatible computers became so ubiquitous and affordable that all walks of society could partake of them. Similarly, reasonably common use cases of ICT like installing software, configuring basic networking, producing media and documents became much easier since technology stabilised, documentation got written and the social intellect of the general population caught up. At the same time the new wave of DIY technologies – physical computing,[30] (Igoe and O'Sullivan 2004) computer aided manufacturing,[31] (Söderberg 2014a)]and synthetic biology[32] (Delfanti 2013) have grown relatively capital intensive – a development hacklabs could not follow on their own terms, while political applications of these technologies remained unclear despite "revolutionary" discourses around them. (Gershenfeld 2005; Anderson 2014; Troxler and Maxigas 2014) These technologies became the basis of hackerspaces – the new wave of shared machine shops – and subsequent, progressively more and more recuperated genres. At the same time a generational

[29] The dot com boom was a largely North American phenomena, where companies with ".com" in their names were overvalued.

[30] Physical computing uses programmable microcontrollers for interaction with the offline world through sensors and actuators: robotics, home automation, drones and the Internet of Things are examples of its application.

[31] CAM includes numerically controlled cutters (CNCs), laser cutters and most prominently 3D printers of various kinds. CAM equipment is usually complemented by a more conventional machine shop of drills, saws and sanders, etc.

[32] DIY biology and biohacking made huge strands in the last years including the availability of cheap equipment and many inspiring use cases.

shift took place too, wherein the people who participated in hacklabs often found lucrative jobs on the market, often based on their autodidact experience developed within the scene, and moved on to a more middle-class lifestyle of rent, family and activism-as-hobby as opposed to activism-as-lifestyle.

Coupled with these internal reasons which only applied to the specific issues hacklabs were addressing, there were substantial external factors which lead to their demise. In fact my feeling is that the internal obstacles could have been overcome – as shown in contemporary hacklabs – if a meaningful context continued to exist for hacklabish activities. The key historical process was rather the demise of the autonomous movement as a whole, in which hacklabs were but one component. Without recounting the whole trajectory of autonomism, it may be sufficient to recap the major episodes here. As Wright (2002) shows in documentary detail, autonomism started as an Italian answer to the crisis of Marxism-Leninism, based on the three pillars of rereading Marx, conducting workers' inquiry and on-the-ground activism in and out of the factories. Autonomists recognised that capitalism was the most important formative context in which workers have to fight for the revolution, but instead of taking a reactive stance, they looked for ways in which the working class could act independently. The theory of class composition emphasised such a movement where workers could construct a coherent outside of capital within their ranks through self-valorisation, that is, through producing their own consciousness, infrastructure, and ultimately, agency.

During the 1970s – also called the "Years of Lead" (Cuninghame 2005, 78) –, these initiatives grew into the massive *territory of autonomy* comprised of a myriad more or less powerful groups[33], constituting a radical left extra-parliamentary opposition that reached its high water mark in 1977. Autonomia distinguished itself through its theory, which became a recognisable tendency of Marxist and then Post-Marxist thought; its practice, which popularised direct action, squatting and media activism (notably Radio Alice, see Goddard 2011), and the closely interactive relationship between its theory and practice which many observers (like Hardt 1996) found exemplary. As Lotringer and Marazzi (2007) attest, the public organisation of confrontation turned to clandestine armed struggle as activists faced overwhelming repression. A similar development took place in Germany, where the Autonomen movement grew out of antiwar and anti-racist student activism (Schultze and Gross 1997; Geronimo 2012). The next, desperate period was therefore characterised by the activity of paramilitary organisations focusing on urban guerrilla warfare, like the Red Brigades (Brigate Rosse) and Prima Linea in Italy and the R.A.F (Rote Armee Fraktion) in Germany. (Aust 2008; Lotringer and Marazzi 2007) These groups became increasingly isolated even if they initially conceptualised themselves as the vanguard which the masses would follow. Even in this period, squatted social centres were important bases for clandestine insurgent groups. By the 1990s these groups were largely neutralised and the autonomist movement lost its

[33] Autonomia Operaia and Lotta Continua were major ones.

revolutionary fervour.

Hakim Bey's 1991 manifesto "T.A.Z.: The Temporary Autonomous Zone, Ontological Anarchy, Poetic Terrorism" captured the imagination of the next generation of militants, activists and hackers, who retreated from open confrontation with the state and capital but retained the core concept of the autonomy: that the radical left movement can act as an independent historical agent through self-valorisation. (1991) As a result, in this period the autonomist tradition was largely articulated through practices, the most important being the occupations that gave its name to the squatting movement. Hence squatting gradually moved to the heart of the movement, becoming its identity.[34] Temporary Autonomous Zones acted as a heartland where prefigurative politics could be acted out through what George Katsiaficas calls the "decolonisation of everyday life" (1997).

The Internet as an infrastructure and especially hacking as a practice fitted neatly into such a configuration. It is a common trope of modernity that new technology is invested with "revolutionary" meanings, holding the key to a new subjectivity and objectivity, to a new man and a new society, as for instance in the works of Dziga Vertov, picked up by Godard in the 1970s, (Brody 2008) leading Italian autonomist Lazzarato and the Turkish autonomist intellectual Baker (Baker). However, in the 1990s the autonomist theory and practice of prefigurative politics coincided with the hacker experience in a more rigorous sense. The spatial ambiguity of cyberspace between physical and virtual space ("temporary autonomous zone"); the metaphysical-ontological ambiguity of hacking between play and resistance ("ontological anarchy"); the semiotic-performative ambiguity of program code ("poetic terrorism") – condensed into the temporal ambiguity of cyberpunk as a retro-fitted sci-fi imaginary coincided with the autonomist strategy of retreating into the future acted out in the present. The slogan of the alterglobalisation movement of the next decade – "Another World is Possible" – follows the same logic of ontological ambiguity between imagination and reality.

However ambiguous, text-only terminals offered instrumental access to the mediation of social relationships which already mesmerised the Situationists and long muddled the Autonomists, as a kind of super-Saussurian underlying architecture of everyday life.[35] In retrospect, it is easy to see that hacklab participants correctly identified the political potential of ICTs, but underestimated the resources brought in against them by the state and capital once the Internet got their attention. Expropriative anarchism as a resource mobilisation tactic was not cumulative enough to keep up with creeping regulation and the influx of investment, so that recuperation was inevitable. The community-run and self-managed, federated social media of the old Internet protocols (like BBSs, Usenet forums, Internet Relay Chat rooms, Indymedia websites and later blogs) gradually gave space to Web 2.0 with its corporate-run and state-controlled, centralised "walled

[34] Another interpretation is that the squatting movement formed as a separate movement parallel to the autonomous movement – for instance such a situation is described in Owens (2009) 81-83.

[35] Darkveggy stated that as teenager the aesthetics of the command line hit him harder than his former passion, heavy metal music.

gardens" (like Facebook, YouTube and Twitter). Similarly, gated communities were erected in urban areas and squatting has been slowly criminalised, giving way to "anti-squat" companies offering cheap rent to students in derelict houses, while art centres opened in industrial ruins managed by the local government. Dadusc and Dee (2015) tell a typical story:

> Amsterdam City Council's *Broeadplaatsen* (Breeding Places) policy allocated €41 million for subsidising between 1400 and 2000 living/working spaces for artists and cultural entrepreneurs. Many evictions were halted and some squats were legalised and turned into cultural centres. Many squatters found compromises with the owners and the Council, renting and buying for low prices the spaces they already occupied. According not only to academics such as Justust Uitermark but also to many squatters and activists, this policy led to the absorption of parts of the movement into providers of cultural services, which contributed to the image of Amsterdam as a 'creative city' and 'helped to co-opt and to prevent resistance against policies that seek to promote gentrification'.

In effect, squatters largely dragged on or dried up during the first decade of the naughties; but by the second, state and capital invariably got the upper hand in Germany, Netherlands, England – the previous strongholds of the movement, listed in order of their fall. Some of the more powerful occupied social centres (like the EKH in Vienna) managed to become a naturalised part of the urban landscape, yet they have always been but the tip of the iceberg. Criminalisation came quickly and stuck at the heart of the most active scenes, while it stamped out others sweepingly.

Recent years saw the abolishment of 'squatters rights' in the Netherlands (Pruijt 2013) and the criminalisation of squatting residential buildings in the UK [Manjikian2013a; HMG2014a].[36] However, in my view criminalisation was merely the final nail in the coffin of a movement that started to decompose anyway. In effect criminalisation was the practical recognition by the state and capital that the theoretical underpinnings of autonomism withered away, and therefore it was weak enough to be suppressed without prolonged fires in the city centres (which was not true in the 1970s at least in the core cities of the Italian Autonomia and the German Autonomen). For instance Dadusc and ETC write that "squatters have had to engage with the languages spoken against them by external actors, instead of telling their own story." This happens because the movement was too weak to be able to impose their own narrative. *In the greater scheme of*

[36] "The introduction of the offence of 'squatting in a residential building' in section 144 of the Legal Aid, Sentencing and Punishment in of Offenders Act 2012 (LASPOA) marked an important turning-point in the UK state's relationship with practices of unlawful occupation." (O'Mahony, O'Mahony, and Hickey 2015, 1) albeit in the same volume Deanna Dadusc and ETC Dee notes that "Squatting is already a criminal offence [since many years] in Scotland and Northern Ireland, so we cannot talk of the United Kingdom." (2015, 109)

things, resisting evictions was fighting for the revolution in the 1970s, defending an alternative form of life in the 1990s, and protecting some political projects in this decade. Such thesis points to three simultaneous shifts: 1. concretisation of the context from a historical horizon to a concrete building; 2. deradicalisation of tactics from urban guerrilla warfare to symbolic demonstration; 3. retreat in terms of realistic ambitions from revolutionary confrontation through building alternatives to preservationism.

Interestingly, the only European area holding on to the practice is Catalonia, where squatting has always been illegal and only sustained by a vibrant movement with a broad vision and a long history. The recent demolition attempt of a major squatted social centre (Centre Social Autogestionat Can Vies) resulted in prolonged street protest and direct action in the neighbourhood after which the authorities were forced to look for a compromise (Local/AFP 2014). Yet, the Iberian hackmeetings which constituted the heartbeat of the political hacker scene (Ferrer 2014) see participation dwindling[37], an increasingly nostalgic air, and few truly active hacklabs on the peninsula, if any.

Taking these factors into consideration, it is possible to evaluate why radical server collectives[38] were pretty much the only component of the hacklabs milieu which survived to this day. While personal computers and Internet access became affordable to the general population certainly by the second decade of the 21st century, online services like email addresses, blogs, website hosting, and virtual servers became virtually free. At the same time all these ICTs – from computers to blogs – became so user friendly that physical association by bodies of users was not bringing comparable benefits for the exploitation of these goods. Logically this would have rendered radical server collectives obsolete in the same way it rendered hacklabs obsolete. However, the media monopolies which have risen on the Internet turned to a business model of capitalising on the data of users, selling it as intelligence products (mainly for advertising) and sharing it freely with law enforcement agencies (often in conjunction with investigations against activists). Therefore, the trusted services offered by radical server collectives continued to be as relevant as ever, if not more. Today, using services of radical technology collectives can count as "conspirative behaviour" (Directa.cat 2014) – showing that they are a variable to be counted with in the power struggles between authorities and social movements. Although it appears slightly irrelevant to the analysis of the collaborative production of open hardware from our current vantage point, radical servers will be an important *shadow* (e.g. control group) in the third case study on how hackerspace participants use Raspberry PI single board computers to develop a practical critique of online media monopolies as well as enterprise server solutions.

In the final analysis, one can say that the core activities of hacklabs were putting

[37] In my rough estimation, 120 at Calafou in 2014; 60 in Bilbao in 2014; and 30 in Marinaleda in 2015.

[38] Like the ones listed by the Italian Autistici/Inventati group in https://help.riseup.net/en/security/resources/radical-servers .

together computer hardware, using and developing free software and setting up sophisticated networks and services. It will be apparent from the next section that all these survived in some form or another into the hackerspaces era, but the latter focus on hardware hacking and creating physical things – typically small scale custom-built electronic artefacts. While hacklabs were sites of incredible technological creativity and productivity, open hardware as a widespread social practice did not exist at that time. Why hacklabs are important for consideration here is the embodied social dynamics they created around the politics of technology. Looking at the collaborative production of open hardware, hacklabs were the first sites of collaborative, grassroots production spaces for ICTs which existed as a widespread, consistent social practice. The addition of open hardware to such a repertoire, along with changes in the social context as described above, changed the quality of interactions considerably. But this is another story to tell in the next section…

6.2 From hacklabs to hackerspaces: Framing technology and politics

6.2.1 From hacklabs to hackerspaces

At the moment the terms “hacklab” and “hackerspace” are used largely synonymously, a tendency which I am trying to reverse through unearthing and documenting their genealogical continuities and discontinuities. Therefore – contrary to the prevailing categorisation – I use hacklabs in their older (1990s) historical sense. This is not meant to be linguistic nitpicking or erudite etymologising but meant to allow a more nuanced understanding of the environments and practices under consideration. A method which I call *digital archaeology* can bring back evidence of historical usage, reconstructing the history of ideas in digital discourse. The evolving meaning of these terms is recorded on Wikipedia, where the loss of historical memory is well documented. Since Wikipedia is a major peer production project and enthusiastically embraced by hackers as a reference point about their own culture, it is not surprising that changes follow the trajectory of the scene. The *Hacklab* article was created in 2006 (Wikipedia contributors 2010a), the *Hackerspace* article in 2008 (Wikipedia contributors 2014b). In 2010, the content of the Hacklab article was merged into the Hackerspaces article. This merger was based on the rationale given on the corresponding discussion page (Wikipedia contributors 2010b). A user by the name “Anarkitekt” wrote that “I’ve never heard or read anything implying that there is an ideological difference between the terms hackerspace and hacklab” (Ibid.). Thus the treatment of the topic by Wikipedians supports my claim that the proliferation of hackerspaces went hand in hand with the forgetting of hacklabs.

Another document on the transition between hacklabs and hackerspaces has been obtained due to the compactness of the rich Dutch hacklab and hackerspace scene when Fish_ contacted me reading an article I published on the topic, and filled me in on some of the details around the foundation of the first hackerspace in the country (2014-11-01). Combined on the relevant pages on their website, we get a particularly close account on the change in shared machine shops which is worth looking at in detail. The history of the space is recounted as follows:

> In December 2008 Fish_ had a visit from his beloving friend Eric Michaud (a serial hackerspace founder and lockpicking expert living in the United States – maxigas), Eric inspired Fish_ to start a hackerspace in Utrecht since there wasn't one yet. The first thing Fish_ did was to contact Kahits and brainstorm how to start a space in the Netherlands. Within a few weeks also XOR was added to the team and this is how the first board of the Randomdata foundation was formed. In the beginning we had no space, we started to do "homespace sessions" and had a lot of fun. People contacted us and soon we had a core group who formed Randomdata. After HAR2009 (the seminal Dutch hack camp – maxigas) we had a group big enough to rent a physical space and we went hunting for a place. In October 2009 we found our space where we still are. It's not big but we have an awesome place to Hack, Make, thinker and a lot of other things we like to do (Randomdata 2013).

There are many things to notice in the text in comparison to the hacklabs story which we already know. First, about the dissemination of the practice we can note three things. One, that the hackerspaces idea first arrived from the USA to the Netherlands, most probably "retrofitted" from Germany, since it was quipped there by Ohlig and Weiler (2007) and we know that hackers from across the Atlantic were touring European spaces and gatherings that year. This speaks to my thesis that the American hacker scene provided a kind of lingua franca through which linguistically isolated scenes came together. Two, that despite the Internet being in full swing by now and hackers no doubt digging it, the "inspiration" travelled on two feet and was transmitted through personal friendships. This highlights the importance of offline contacts and personal ties for a scene which lives in the popular imagination as a stereotypical virtual community of people hiding behind pseudonyms. Three, that the critical mass was achieved at a major hacker gathering which is part of the heartbeat of the scene as it has been argued. We will see that there are other sources too which point to the fact during this particular hacker gathering participants arrived to the understanding that they should be founding hackerspaces when it is over. For instance the second pioneering hackerspace (RevSpace at The Hague) was also founded shortly after the HAR2009 and others followed suit. This dynamics recalls the above reference to the hackmeetings in general and to the one in Milan 2003 in particular.

Second, there is no reference to other than the hacking community neither in terms of inspiration nor in terms of resource mobilisation, as in Trocchi's recollection earlier ("the imaginative space that was opening in the young and angry minds of communities of squatters, activist and ravers"). On the one hand this underlines my argument that on contrast to hacklabs, hackerspaces are not embedded in a broader political or cultural movement, but grow out of the hacker scene directly. On the other hand it highlights the point made further on that renting is an uncontroversial move in hackerspaces circles which alters the political economic basis of the organisation and therefore its engineering culture too. It is notable that at that time Utrecht still had a major occupied social centre (Ubica, evicted in 2014) and other squats, so at least in theory occupation as an option has been available.

Third, in the description of activities there is no reference to a lay audience or even a democratic ideal. *Access activism*, as I termed hacklab work, is not mentioned, even though there is a clear intention to create a shared space open for qualified members to provision the necessary tools, people and ambience for unalienated labour on technological artefacts. In fact the actual definition of the space places the same clues (Ibid.):

> Randomdata is group of people who want to do "technical" stuff. Randomdata is a hackerspace. A hackerspace is a real (as opposed to virtual) place where people with common interests, usually in science, technology, or digital or electronic art can meet, socialise and collaborate. A hackerspace can be viewed as an open community lab, workbench, machine shop, workshop and/or studio where people of diverse backgrounds can come together to share resources and knowledge to build/make things. We build, we break, we create, we invent and of course we hack. There are no actual limits to creativity (unless it's illegal ;)).

Coherent with the renting solution, the last phrase makes it clear that illegality is off limits, at least in the public presentation of the shared machine shop. In fact the page continues stating that "Randomdata community is supported by the Randomdata Foundation (Stichting Randomdata)" which is another step towards institutionalisation hence adopted by the majority of hackerspaces. Now the hackers play it safe, legit, and according to the rules, without the illusion that they can create their own "Temporary Autonomous Zones" or cyberpunk virtual realities. At the same time, as we shall see, the recognition of the theoretical notion that technology is deeply continuous with the capitalist world and the practical realisation that the social movements are not strong enough to build their own world single-handedly open up new possibilities for an alternative relation to technology in the confines of shared machine shops.

In agreement with the above, Fish_ wrote that "the biggest change was that the spaces were changing from a squatted space to a space with a landlord … to arrange more stability". Turns out that the switch from the hacklab approach to the hackerspace approach[39] was discussed explicitly by the Randomdata founders,

[39] Both established genres internationally by then.

who found that both of the previous major hacklabs – PUSCII and ASCII – folded their physical activities after a few years because participants were tired to move from space to space at unpredictable intervals. In fact these discussions were very topical indeed since in 2007 when this happened, the local Utrecht hacklab called PUSCII lost its last physical location at the Stationsplein and the year before the hacklab in the capital (ASCII in Amsterdam) terminated too. However, Randomdata participants had concerns about regulation, institutionalisation and elitism, which were expressed in the strong desire to make the board of the Foundation subordinate to the members in properly democratic ways. As the quotes above attest, the model worked out in practice and ensured both the longevity of the project and the ability to accumulate more equipment – essential for the transition to hardware hacking, robotics, and electronically controlled manufacturing which are at the core of typical hackerspace (but not hacklab) activities.

6.2.2 Framing technology and politics

In summary, the relationship between technology and politics is constructed in diametrically opposed ways in hacklabs and hackerspaces.

Hacklabs are conceptualised as explicitly political organisations since they are embedded in a social movement which questions not only intellectual but also private property (personal communication, Lunar 2013-04-24). The devotion to FLOSS (Free, Libre, Open Source Software) is continuous between the two types of shared machine shops, as well as the promotion of alternative licences for the production of free culture (like Creative Commons) – and both of these legal techniques can be understood as a subversive critique of intellectual property. However, occupying buildings extends such critique to private property in general, and the expropriation of empty buildings points to the critique of a specific form of property: capital. For this reason it is not unusual for hacklab participants to engage in direct action against the state and capital even if it is not technology or culture is at stake, but for example solidarity with other social groups. Therefore hacklabs exhibit more coherence in their approach to property politics, and do not confine their concerns to the realm of engineering. **A hacklab participant is first and foremost a politically engaged person, who acts on his conviction through her specialisation.**

Hackerspace members on the other hand owe their loyalties primarily to the hacker scene (an engineering culture), defending the values and interests of that specific social group, mainly connected to user control over technology (including privacy, anonymity, open data, free technologies, etc.). They question intellectual property through the critique of copyright and the development of free software and hardware like hacklabs but do not go all the way to take action against private property, and don't necessarily recognise the problem with capital. Hence it is allowed for instance to create spin off companies from hackerspace inventions while even cashing in on your skills in multinational corporations is looked down upon amongst hacklab alumni. It is not that the hackerspace scene would not mobilise around political issues sometimes, even engaging in street protests like in the case of the campaign against ACTA (Anti-Counterfeit Trade Agreement). It is that such engagement is confined to the realm of a professional ethics, all the stronger for this. **A hackerspace member is first and foremost an engineer,[40] who may engage in politics to defend her idea of technology.**

Of course both groups include many hobbyists, lifestyleists and tinkerers who seek to stay away from anything which resembles political action. In a way shared machine shops in all their manifestations provide a shelter for self-centred or technology-centred individuals – arguably adherents of commodity fetishism – who do not want to be distracted from engineering neither by bottom-up (social movement) or top-down (state and capital) pressures, but instead believe in "The Right Thing" or *pure technique*. Nevertheless, even they shape technological possibilities and therefore the social dispositif therein, which is in itself a political activity. It goes without saying that their broader milieu has a profoundly effect on their ideas of what counts as a correct implementation.

There are also a few caveats to such analysis however.

(1.) Hackerspaces may not be as big a step away after all from hacklabs as the analysis above would suggest. **At their core, the determining dynamics in both forms of shared machine shops is the self-organisation of unalienated labour for collaborative grassroots research and development, or in other words the infrastructure of peer produced expertise organised in a radically participatory way.** This is what sets them apart from similar institutions (see the *Misc. forms* section at the end of the chapter). Ultimately, their most political gesture which they both share is the development of a technique which serves lives directly, e.g. which finds its end, means and invests its meanings in the participants ("workers") themselves.

This encompasses the whole world of craftsmanship (more on that in the next section) but in the final analysis points towards a concept of engineering as unalienated labour. It just so happens that hacklabs gain their material basis for the realisation of the ideal through expropriative anarchist techniques (e.g. primitive accumulation) exercised by the poor, while hackerspaces gain their material basis for the realisation of the ideal through the exploitation of classic middle class privileges (e.g. leisure time and surplus income to be spent on hobbies). These political economic factors have a considerable effect on the engineering cultures which develop in the respective spaces, yet the yearning for unalienated labour is the same on an anthropological level.

[40] An engineer in practice, though not necessarily with a degree.

In the final analysis, the political potential of both genres of shared machine shops should not be measured by their instrumental contributions to social movements or the subversion of social order, but by the cultivation of an alternative vision of technology[41] that goes beyond the confines of capitalism, liberalism and even modernity. A bottom-up practice of engineering organised outside of the modern institutional grid, sustained through a semi-independent culture, driven by the desire for unalienated labour (Söderberg 2008; Himanen 2001). The difference lies in the consistency with which each genre can put forward such a vision.

(2.) The advantage of the hacklab approach is apparent, e.g. that it seeks to address social problems as primarily political problems (which they are). Participants see themselves in their role as hackers as specialists serving the more comprehensive movement (which they do). So far so good, but in practice the totality of this vision is often coupled with a narrow-minded solidarity and a restricted outreach. Social groups who are not part of the movement per se or one of the target groups of the movement (for instance immigrants) are not catered to. In actual fact the appeal of both the hacklab project and the squatters' milieu is often too "spiky", "edgy" and "dodgy" to accommodate the taste of "normal people", i.e. most social groups comprising the general population. Such effect of radical politics is called the *activist ghetto* and the result is that hacklabs mostly serve the local scene rather than any manifestation of the people at large, or even any recognisable manifestation of the downtrodden.

Hackerspaces are recuperated in the sense that they are a 'tamed' version of hacklabs that conform to the socio-economic order of the state and capital. However, exactly because they are more "mainstream", recuperation means accessibility. Hackerspaces members have less scruples to do what needs to be done to forge necessary alliances with any particular social group, be that neighbours, corporate donors, civil society organisations, let alone squatters or sometimes even the secret services and state authorities. Therefore – as many classical liberal projects in the second half of the twentieth century – they can easily end up more approachable then hacklabs, and more cooperative in their actual practices. Participants generally don't aim for all-encompassing social change, and often unashamedly owe their alliances to their own social group, or do the same under claims of standing for universal values. Fortunately, the interests of hackers in many important ways coincide with the interest of the downtrodden. For the moment this has to be a conjecture here but I will leave it at that, to pick up on it later.

(3.) Finally, as (Söderberg 2013) observes, hackers also hack politics, sometimes with surprising results. Modernity is deeply invested in technological determinism, and capitalist managers have used technological determinism so many times to introduce changes detrimental to workers as apolitical, progressive, and historically necessary changes. Given all the discursive investment in technological determinism, it can be mobilised against its masters as a frame for collective action. Hackers as (self-proclaimed) experts are in a good position to present their

[41] Called *cybernetic ontology* by (Pickering 2010).

political ideas for social reform as purely technical arguments about increasing efficiency. For instance copyright does not have to be questioned politically as the basis of property, thus theft: one merely has to observe that – due to the unstoppable development of technology – in the 21st century copyleft is a more efficient method of distribution. Then hackers have history on their side, or even better: the changes they propose have to be introduced because of simple historical necessity as the next administrative step in technological progress. There may be voices which oppose such changes (such as the media industry) but they are bound to be defeated in due time because they are just archaic. Ironically, by virtue of their apolitical public image hackerspaces can sometimes sport more effective strategies for the intervention in political processes than hacklabs.

Thus – to summarise rather schematically – the hacklab activities arguably revolve around the desire for a widely conceived *political technology*, while hackerspaces pursue a more focused *techno-politics*: on one side technology is framed by politics, on the other technology frames politics. The tragedy of hacklabs is that they lack the resources to say what they have to say; the comedy of the hackerspaces is that they have all the resources but lack words. Similarly to cultural studies scholars in the 1970s, in such a historical situation it is especially important to look at cultural artefacts and their socio-political meanings.[42]

6.3 Hackerspaces

Hacklabs form the historical background, but hackerspaces are the actual research sites of this study, and therefore they warrant a closer examination than hacklabs. Hackerspaces are veritable hacker clubs run on membership fees by the hackers, for the hackers. Anyone with a technological interest is welcome to meet, socialise and work on projects in hackerspaces, which provide an ideal environment for such activities. In practice this means a machine shop, electronics lab and network connectivity on the one hand; a bar, sofas, and chill-out room on the other. Perhaps they are best described by the slogan of Hackerspace Singapore: “tech scene’s community centre.”[43] *Peer production in hackerspaces happens through a more or less informally organised infrastructure facilitating technically minded individuals and groups to share essential resources like knowledge, time, space and tools.*

[42] “The rise of cultural studies itself was based on the decline of the prominence of fundamental class-versus-class politics.” (Lash 2007, 68–9) – in the same line I suggest that looking for the politics of small scale electronic artefacts from an STS point of view is only necessary – or rather the only option – during an ebb in large scale political mobilisations.

[43] http://hackerspace.sg/

Hackerspaces can be considered to be at the height of their popularity at the moment, and widely emulated by subsequent genres of *shared machine shops*, the full spectrum of which is considered in the last part of this section. The website hackerspaces.org holds a database of more than thousand self-reported active hackerspaces as of January 2015.[44] While the above description would fit most, this dissertation focuses on North European hackerspaces simply because they have achieved the highest level of consistency in terms of practices. These number around 150, and I have surveyed on-site roughly a fourth of them (see the *Methods* chapter for more details).

The three determining factors I find which position hackerspaces in the ecology of shared machine shops are the following – notice that they all point to internal contradiction that drive the dynamics of these communities:

(1.) *Explicitly apolitical and unaffiliated projects*: Hackerspaces never represent themselves as politically partisan, and stand apart from academic, non-profit, public or private institutions, allegedly in order to be able to serve all types of potential members openly: they take a position of neutrality. Of course rejecting affiliation with any modern organisation is itself a political stance, and apolitical policies mean that the bottom line of mainstream society (e.g. liberalism) and the bottom line of the hacker scene at large (e.g. libertarianism) will prevail. At the same time, each hackerspace has its activists and anarchists who debate and organise quite openly, in a way exploiting the neutrality of the hackerspace in two ways. On the one hand the hackerspace is a safe place to discuss and practice politics because its immaculate neutrality makes inconspicuous and hard to target by authorities, especially that is is (see the next point) a perfectly legal organisation. On the other hand neutrality does mean that most members are not from "the activist ghetto", so that agitators are not merely "preaching to the choir". In the London Hackspace I had the fortune to witness the three rooms filled in with the edges of the membership: in one, Occupy activists have been mending their wiki; in the middle mind hackers were busy at work with hypnosis; in the third a defence contractor was explaining the new CB radio system of the local police.

(2.) *Institutionalised spaces with an anti-institutional ethos*: The legal implementation of hackerspaces include foundation and associations, which have a more or less serious role to play in the life of the hackerspace. In many spaces the foundation is simply a legal facade to council communist self-management practices, whose only role is to accept donations, hold the contract and pay the rent. In other spaces the board of directors have full authority to exercise their roles, holding closed meetings, deciding on the investments, savings and new acquisitions, or excommunicating members from the hackerspace at will. In most spaces the social dynamics balances between these two extremes. The power of the "shop floor", however, is guaranteed in many ways, of which three are worth to mention. Firstly, hackerspaces are officially there to serve their members, therefore board members are formally required to cater for their needs and take

[44] http://hackerspaces.org/wiki/List_of_hackerspaces

into account their opinions. Secondly, the classic dynamics of peer production projects apply, e.g. that even if there is a "benevolent dictator for life", there is no chain of command which would bind the members to obedience, and the project is only alive as soon as members are happy to participate. Thirdly, hacker culture itself, of which hackerspaces are perhaps the most vibrant contemporary manifestations, nurtures a decidedly anti-authoritarian ethos, which leaves its mark in the organisational culture of hackerspaces.[45] Such dynamics is different from more institutional "next wave" shared machine shops, which are usually affiliated with academia or funded by the organs of state or capital – as long as their sustainability of a shared machine shop is not solely in the hands of its members, there are many political-economic structures which come into play in deciding the fate of the organisation. As seen in the story of Randomdata earlier, hackerspaces institutionalised hacklabs in order to strengthen hacker culture – for better or worse. However, the shared machine shops of the next generations more often than not aim explicitly to instrumentalise hacker culture for their various goals.

(3.) *Highly accessible rented spaces*: as Fish_ acknowledged, the political economy of hackerspaces is structured by the fact that they are situated in rented properties. The reliance on a core of middle class membership affluent enough to pay the rent, the requirements for a legal organisation behind the community, the facade of neutrality are all pretty straightforward consequences of this resource mobilisation strategy. However, as usual, hackers are pushing the boundaries of what is possible to do in such a framework. The ground rule of hackerspace operation is nonstop access to the space, which already rules out many otherwise suitable properties. Next, as the second case study amply shows, hackers spend quite some time transforming their built environment, which may involve pulling down walls and building new ones, installing shelves and running copious amount of cables across the rooms. Not all landlords appreciate hands-on attitude like that and some hackerspaces work around limitations by devising ways to place sensors, cables, etc. in a non-invasive way. For instance in the hackerspace in Augsburg the door system includes a contraption which sits on the door handle and opens the door robotically when provided with the right credentials (Personal communication, gamambel, 2015-01-24).

Hackerspaces therefore occupy a liminal position where their mission is not overcoded by the mission of any wider political project (as in the case of hacklabs) or the mission of larger institutions (as in the case of most next generation shared

[45] Participants of other genres of shared machine workshops recognise their hacker heritage, but their representatives often state that they purge the tropes of hackerdom from their vocabulary because they don't want to scare away potential members, collaborators and partners with the bad public image of hackers. While this is plausible indeed, another reasons could be to curtail the "rebel spirit" of hacker culture in these more institutionalised spaces. Ironically, the edgy image of hackers are exploited by institutions for the same reason, for instance when the Brazilian Parliament establishes a "hacker space" inside the building where citizens work on open data, open government, and similar topics. (Swislow 2014) While a laudable in itself, it is easy to see from the description here that such a data laboratory is far from the hackerspaces ethos, even if inspired by hackerspaces.

machine shops) – they are primarily governed by the internal dynamics of peer production processes as they are articulated in hacker culture in general. As I repeatedly argued, this is what makes them privileged sites for the study of peer production processes, as — following the argument of Dickel, Ferdinand, and Petschow (2014) – they form a protected niche of experimentation with peer production processes and small scale open hardware production, free from pressures political or economical. As we shall see later on, such situation is similar to the early days of FLOSS or even the Internet where much of the technology was developed without concise pressure from more powerful social actors. Even if that gradually changed in FLOSS development as it became a key part of the new economy, hackerspaces are directly influenced by the culture of FLOSS development since their members are often active coders who contribute to FLOSS. However, since FLOSS as a case of peer production has been extensively described in *Chapter 4*, it is kept in the background here.

The essential point is that hackers are not simply hacking or peer producing software and hardware in the hackerspaces: as Grenzfurthner and Schneider (2009) first noted, they are "hacking the spaces", e.g. experimenting with organising peer production itself.

6.3.1 History

6.3.1.1 Streams of hacker culture Hacklabs and hackerspaces are both squarely rooted in hacker culture, yet hackerspaces maintain and nurture a more intimate connection to the core processes of hacking. Another way to put it is that hackerspaces have little other cultural heritage to mobilise than hackerdom: as suggested earlier, hacklabs stemmed from autonomous politics in general and media activism in particular, along with hacker culture. In fact it would be worth a research line of its own how hacklab participants misunderstood the mainstream of hacker culture, which resulted in one of the most productive encounters and mutations of hacking, but also resulted in a lot of disappointments and frustrations. Hackerspaces position themselves at the centre of the field, and therefore don't encounter such challenges. Therefore it is more appropriate to treat the history of hacker culture here – in the hackerspaces section – than in the hacklabs parts.

Hacker culture first developed in the United States, which later served as a common ground for further developments in various directions. As noted earlier, the US hacking experience – both as a mythology of its "classic" era and as an ongoing narrative with its twists and turns – served as a common ground of knowledge and as a medium of communication (establishing a common language) between various developments in Europe. The famous figures and leading institutions of US hackers have been known and respected by hackers of all kinds in the continent, even when those hackers worked isolated from each other. Practices of hacking usually moved from the States to Europe, in the same way that in Europe they tended to move from Italy to Spain in the South, or from Germany to the Netherlands in the North. However, as we will see in the next section, hackerspaces are an important and significant counterexample to the general tendency. In the case of hackerspaces, it is very clear that North American hackers came to Europe to pick up the nascent idea and the experiences of pioneer projects here, and allied with their continental peers to bring about the golden age of hackerspaces on both continents.

The principal source – indeed, probably the founding epic – of hacking history is Levy (1984). It is an important document for three reasons. Firstly, it is a well-written quasi-ethnographic treatment of some important sites of emergence. Secondly, it is widely read by practitioners who derive their sense of belonging and identity from its pages. Thirdly, it gives the canonical definition of the famous *hacker ethic* as a kind of conceptual summary of the empirical material in the book. Therefore it is far to say that it succeed not only to document but to channel hacker culture, lending it credence as a primary and secondary source too. For these reasons the recapitulation follows its strides even if it means perpetuating myths rather than pointing out under-appreciated historical developments. About the latter, it should be noted that for instance the early days of hacking in other countries is scarcely documented in the scholarly literature.

For reasons of presentation the origins of hacker culture can be roughly divided into three streams: *academia*, epitomised by the MIT; *civic hacking*, epitomised by the Homebrew Computer Club and the *underground*, epitomised by the phreaking scene. To imagine a tradition founded by researchers, hobbyists and criminals gives a high fidelity impression of the overall tone of hacker culture. The hacker habitus can be analysed as a triangulation of these three. In retrospect the most prominent contributions of these streams were perhaps in the following areas: the *academia*, software; *civic hacking*, hardware; the *underground*, networks. What tied the three steams together was building a knowledge commons, e.g. the collaborative production of knowledge. “Information wants to be free” – an obvious case of hard core technological determinism – was one of the first slogans of the scene, and sharing technical ideas the principal purpose of hackers talking to each other.

Not incidentally, hacker culture – just like autonomous politics — stem from the “cultural shock” around 1968 (Wallerstein 2004, 16–17), which explains many of its idiosyncrasies. Firstly, it bears all the traits of a youth subculture: both iconoclast and symbolically rich, infinitely productive but hanging out lazy.[46] Secondly, it can be interpreted – as I did earlier – as part of a generational retreat after the revolutionary fervour of that moment into a professional ethics which works within the confines of specialisation. Thirdly, it does bear the marks of cosmopolitanism made possible by many-to-many communications through electronic networks developed under the supervision of hackers. Let us survey the three streams briefly, make the connections to the hacklabs and hackerspaces of latter day and move on to the actual history of hacker clubs.

[46] It is worth to note that youth culture as such did not really exist before.

Following the Second World War the *academia* enjoyed three incentives. Once, the wartime innovations could be developed in the time of peace into civilian applications. Twice, the Cold War brought ample funding for base research into open fields. Thrice, the first generation of young people empowered by youth culture found its most receptive environment in the university. As a result, young researchers had a lot of freedom working on amazing challenges. Cold war doctrine advocated a strategy of economic growth through technological advances which would be shared between the military, the academia and the private sector, raising the quality of life at home and ensuring military superiority abroad. Such strategic openness may be seen as one of the ideological basis for the values of sharing amongst hackers. Barbrook (2007) identifies nuclear reaction, rockets and artificial intelligence as the three key technologies in the social imaginary of this era, and of course the lot of hackers was to realise the latter. The Artificial Intelligence laboratory at MIT, along with a string of similar institutions at other universities as well as private institutions like Bell Labs were at the forefront of these efforts. In military-industrial complex, military funding and technology transfer were the defining traits of the ecosystem, so as far as technologies like computers, networks and algorithmic processing were concerned, the state, academia private sector were permeable, and given the early stage of development, there were good reasons for the culture of the academia to prevail. Where conflicts ensued, hackers found creative solutions: for instance the birth of free software (1989) can be interpreted as a conflict between corporate and academic practices of sharing research results (see *Chapter 4*). Even if AI was never realised, the era gave hackers the Unix operating system technically, and the Jargon File (Steele and Raymond 1996) culturally. The former is still the basis of Linux (powering embedded devices, servers and Android smartphones) and OS X (powering Apple computers and mobiles), while the latter is still quoted on hackerspace websites and conversations.

The Homebrew Computer Club (Levy 1984 Chapter Two / 10.) was perhaps the most prominent representative of *civic hacking* where "amateurs" — in reality often accomplished engineers with formal training, just like in hackerspaces – who could not wait for the market to deliver the goods but took innovation into their own hands and built microcomputers for everyday users. As Levy recounts, the division between engineers and activists have existed amongst them since the very first meetings: some simply wanted computers for themselves (and by extension for the masses) while others wanted a better society (through spreading computers). Interestingly, both the former (most famously Steve Wozniak) and the latter (most famously Lee Felsenstein[47]) formed microcomputer companies soon. Perhaps they came closest to the hackerspaces of today, sharing expertise, ideas and even hardware amongst each other through volunteer run information systems connected through telephone (Bulletin Board Systems) and physical meetings (like the fortnightly ones of the Homebrew Computer Club). Along with the ethos of sharing, middle class privileges to afford - in terms of time

[47] Later in 2009 the "Founding Sensei" of the Hacker Dojo hackerspace in Mountain View, California.

and money – a “hobby” and the market availability of microchips – the heart of the microcomputer – have been amongst the enabling factors. Without these “amateurs”, neither the entrepreneurial spirit of startup companies, neither the personal computers of today would not exist.

The *underground* was an extension of the *phreaker scene*, groups of hackers who researched telephone networks. The principal source here is Sterling (1992) rather than Levy (1984) mainly because of the obscurity of the subject, its later maturity and it subsequent rise to the limelight (following FBI / Secret Service actions like Operation Sundevil). Before the break up of the Bell Telephone Company (affectionately called Ma Bell by hackers), the telephone systems were the largest public networks available. It was within the norms of the scene to gain “unauthorised access” to these networks and explore their innards, feeling around and testing what can be done with them, and perhaps even providing some necessary services for the research community. However, doing damage or disrupting operations were not acceptable, except as revenge. Indeed, while information was shared freely between phreaker groups (as in the other two streams) at first, many realised later on that knowledge can get “into the wrong hands” so that a more nuanced approach is necessary: the legendary groups of the era – Legion of Doom and the Masters of Deception (Slatalla and Quittner 1995) – had informal circles of initiation where trusted members gained access to more and more arcane knowledge. Of course the underground included criminal elements who took advantage of the knowledge therein, and even well known hacker like John Draper or Wozniak were known to build “blue boxes” for making free phone calls. The field of *information security* was established by these groups: indeed, their legendary undertakings were in part only possible because no such a field existed.[48] The use of pseudonyms to protect identities and the idea of hackers banding to groups and gangs originate from this stream as well as the field of information security, a type of “independent research”.

Interestingly, it seems that the stream with access to the most resources – *the academia* – was most obsessed with the least resource intensive area (software); the stream with modest resources – the *the hobbyists* – with a moderate one (hardware); and the security researchers with little else then time on their hand – *the underground* – with the resource heavy infrastructure. It is hard to say why, but it proves that motivation, free time and unalienated labour can go a long way even in the absence of political authority and economical capital. This is an assumption without which no hackerspace is founded. From the above we can conclude that the cultural and technical baggage hackers bring to hackerspaces includes a strong disposition for sharing knowledge, “independent research”,

[48] Culprit (1987) reports on an older system that “it is quite easy to hack into a Prime running a version 18 of Primos. The external security is rather poor. All you need is an ID to log on. There is no password prompt, thus getting an operator’s account is rather easy.”

working in groups, using pseudonyms, Unix systems (like Linuxen and BSDs), exploring networks and most importantly for us: building hardware.

In particular, we can say that the *access activism* of hacklabs is closely associated to the *underground* stream of independent research and free phone calls, exploring larger infrastructures on the verge of criminality.[49] On the other hand hackerspaces are more closely related to the *civic* hacking stream – not only because of hardware hacking specifically but since they are oriented towards bringing the priced products of the military-industrial complex to the masses through simple, understandable and affordable prototypes and startup companies. Finally, perhaps it is not far-fetched to claim that the new wave of shared machine shops like Fab Labs can be situated in the tradition of *academic* hacking, since they started at MIT's Centre for Bits and Atoms and new ones are commonly associated with universities.

Coleman and Golub (2008) coined the idea that repression has been instrumental in the organisation and perhaps institutionalisation of the hacker scene. Indeed, the early 1990s brought hackers into the attention of the authorities and hackers were forced to defend themselves not just technically but politically: thus some of the key institutions has been established which mediate between members of the hacker scene internally but also between the scene and its publics externally.

The most prominent example is the Electronic Frontier Foundation, founded by a curious Californian combination of John Gilmore, FLOSS developer (remembered as the maintainer of various GNU tools like the Debugger); John Perry Barlow, poet and writer (remembered as lyricist for the Grateful Dead) and Mitch Kapor, the entrepreneur (Lotus Development Corporation).[50] The original incentive was to organise the defence of hackers in some high profile cases of police repression, like the aforementioned Operation Sundevil, a coordinated crackdown on underground hackers across the country; and the raid of Steve Jackson Games in connection with the manuscripts of the GURPS Cyberpunk role-playing game. The mission was to "raise and disburse funds for education, lobbying, and litigation in the areas relating to digital speech and the extension of the Constitution into Cyberspace." (Denning 1996, 157; Barlow 1996a, 486) The founders felt that society in general and law practitioners in particular were dangerously (for the hackers, that is) uninformed about emerging ICTs, the Internet and hacking. Barlow later developed these impressions into a full fledged *Declaration of the Independence of Cyberspace* (1996b) modelled on the Declaration of Independence, speaking from the position of the Founding Fathers about an Internet usurped. EFF as an organisation continues to define the

[49] Old school underground phreakers/hackers also practiced *trashing*: searching the refuse of selected interesting organisations for intel and equipment.

[50] Seed funding was provided by Kapor, Wozniak of Apple fame and an Anonymous benefactor. (Wikipedia 2015)

political engagements of hackers until the present day, and the declaration itself had profound impact on the generation which read it as much as for instance contemporary info-activists like Anonymous.

The first case was the ongoing ordeal of the hacker Kevin Mitnick, who Coleman and Golub (2008) quote addressing the crowd of the first HOPE[51] conference in New York (1994), organised as part of the campaign against his prosecution: "I was the guy pinned up on the cross." HOPE became the backbone of the North American hacker conference circuit through annual iterations, and now an important meeting place of hackerspaces participants. The authors state that:

> Starting in the mid-1990s and continuing until Mitnick's release in January 2002, the Free Kevin movement schooled the hacker underground in new political idioms and activities. The hacker underground supplemented its politics of transgression with traditional forms of political protest that were more public and organized than any that had come before.

The continuity with the phreaker scene is evident: the location is Hotel Pennsylvania in New York which has the longest phone number in New York in continuous use[52] – therefore hackers were also active in the campaign to save the building from demolition and the hotel from ceasing operation.

In Europe institutionalisation was also guided by repression. Patrice Riemens who was involved involved in the legendary hacker group around the Hack-Tic magazine (sometimes called Hippies from Hell) in the Netherlands told me that facing increasing threats from authorities led them to picture themselves as activists rather than hackers, to clean the image of the underground hacker by turning it into a brand of social activism for the public good. Projects like the Amsterdam Digital City (De Digitale Stadt, aka DSS) were initiated with much fanfare and government support, achieving such a popularity that "Within a week of DDS's inauguration, no modem could be obtained in Amsterdam for love or money." (Tan 1995):

> The Amsterdam Municipality decided to subsidize the experiment, together with the (national) ministry of economic affairs and that of the interior. For the first time, DDS enabled Amsterdammers to look on-line into the council's minutes, to consult official policy papers and to request information from the digital town hall. But there were other activities, too.

According to this testimony, the collaboration between the hacker underground and the state organs looks perfect, delivering much needed information services, self-organisation support and democratic participation experiments to the citizens.

[51] Hackers On Planet Earth

[52] PEnnsylvania 6-5000, (212-736-5000).

Complementing Tan (1995)'s report, the two studies a decade apart by Besselaar and Koizumi (2003) and Beckers and Besselaar (2014) evaluate the project in a more historically and sociologically informed context. All agree that the initiative of the Hippies was an interesting experiment in collaboration between the nascent hacker scene and the institutions, in sharp contrast to the general autonomist and particularly squatters' strategy of positioning grassroots movements clearly on the other side of the barricade when it comes to the organs of state and capital. Teffer (2014) reports that Tjarda de Haan, the first official web archaeologist at the Amsterdam Museum is working on archiving and preserving the system which is now the first officially recognised digital heritage site. Typical for both institutional propaganda and grassroots initiatives, sources agree that despite hopes the system did not actually deliver democracy to citizens, but provided a rich and inspiring first experience with what came to be called Smart Cities initiatives. Discussing the roots of the Dutch hacker movement in alternative (e.g. autonomous) culture, the renowned cyberculture connoisseur Patrice Riemens[53] comments that:

> Next to this alternative movement, there was also that of the Dutch hackers who had the good sense to quickly dispense of the outlaw or cyber-terrorist label and were recognised as a social movement (Casalegno 1999).

In this vein Hack-Tic magazine – again, famous for its publicising independent research on how to actually hack and break all kinds of security systems[54] – ceased publication, as far as I could find out mainly because of police pressure. Between the beginning of the Digital City and the end of Hack-Tic several things happened. On the Digital City side, the infrastructural initiative of Hacktic Netwerk slowly became (1991-1993) the pioneering Internet Service Provider company XS4ALL. Arguably, collaboration with the state went hand in hand with the collaboration with capital. On the Hack-Tic side, the first hacker meetings replaced the zine (1991 and 1993). Interestingly, the conclusion from both sides of the Atlantic was that the controversial points of independent research bring less repression if discussed at open face-to-face meetings offline, then if they are published in writing in the underground press. Nonetheless, the Dutch hacker camps continued to suffer from infiltration or sometimes even complete repression. (Maxigas 2014b) Finally, in 2000 the digital rights foundation Bits of Freedom was set up to support the technical efforts of hackers in the legal and policy arenas, arbitrating between the preferences of the hackers, the policy making of state actors and the market dynamics of capitalists. The latter organisation bears a close resemblance to the EFF in its inspiration.

All in all, I tried to demonstrate here that the 1990s featured a decisive tendency towards institutionalisation in the hacker scene. I made this argument following

[53] Key member of the Hippies from Hell.

[54] "How to copy the data on the magnetic stripe of your bank card?" or "How to build your own pay-TV descrambler".

three threads supported by historical data. Firstly, expanding on the idea of Coleman and Golub (2008), institutionalisation can be understood as closely intertwined with repression. Secondly, internal institutionalisation happened in the form of establishing what I call the circuits of the scene: online forums just as much as meetings of bodies, where symbolic meanings and techno-political strategies can be worked out. Thirdly, external institutionalisation resulted in an array of institutions (EFF, the Tor Project, Inc., XS4ALL, Bits of Freedom, and soon the Hackers Foundation, etc.) which could translate meanings and mediate conflicts between the hacker scene and broader social structures such as the media, authorities or corporate interests.

In the final analysis, a number of things happened at the the end of the 1990s which dispelled various illusions – as moral as metaphysical – about the relationship between hackers, technology and society. I attempt to represent this shift in three historical and logical steps. Once, hacking entered a saturation point in the media externally and an expansion of its adherent networks internally which prepared it for taking a next step in entering the mainstream arena. Twice, material conditions changed when capital at large (at first mistakenly following the lead of Barlow's Wild West rhetorics and the wider exposure of hackers) discovered the Internet as an abstract space for limitless investment. Thrice, the ideological ground shifted as the conceptual borders between the scene and the wider world, cyberspace and reality collapsed. In the following paragraphs I explore these steps in the inverse order.

As the Cold War approached its close, the military-industrial complex which provided the political-economical context of the first period of hacker history suffered a setback. Simultaneously, the Keynesian policies of strong state investment in technology research and development slowly bore fruits and capital needed space for expansion. The rise of neoliberalism facilitated opening up markets for the globalisation of mainly US capital. Cyberspace fitted into such a logic as a space of self-organisation, innovation and growth offering to fulfil the fantasy of limitless growth. Interestingly, the abstraction – the basic movement of capital – that ICTs allowed closely resembled and facilitated the process of financialisation. Foster (2010) traces financialisation through the second half of the twentieth century: even though he does not focus make it the focus of his argument it is clearly shown in Figure 1. that private investment driven by loans increased with unprecedented speed to an unprecedented scale throughout the 1990s, reaching its plateau around the end of the decade. The pattern fits closely to the narrative of the dotcom-boom. While during the first reign of the military-industrial complex the state was the primal actor in political economy, in this period arguably private investment had the upper hand.[55] Hacker culture reoriented accordingly.

[55] In this respect there is not as much difference as it is usually assumed between the socialist block and the "free world" (and as a corollary, neither between the pre-1989 and post-1989

The dotcom boom was a legendary period when the conceptual architecture, ideological visions and the management practices of the startup company solidified. The canonical myth recounts that the startup company begins in a garage where two friends decide to take over the world while tinkering on their original ideas. The company grows with astonishing speed and the founders seek venture capital investment to keep up with developments. "Angel investors" finance the expansion of the company, betting on its success. Most companies suffer a quick death in a few years' time as it turns out that the big idea was not so relevant to real world applications after all, or the implementation plan was based on illusions of a more mundane nature. However, some of them make it to NASDAQ and establish a quasi monopoly in the specialised market that they largely created themselves. At this point the founders either sell the enterprise to a real monopoly that has a grip on a more general portion of the market or try to compete with them. In the fairy tale version a sustainable global corporation is established, typically still headquartered in the United States.

The dotcom boom was based on the recognition of the Internet as a potentially infinite global marketplace, and the idea that any company that has ".com" in their name can grow massive from low seed funding using a relatively small amount of fixed capital. Taken in the middle term of half a century both assumptions were fundamentally correct, but the recognition was missing that the new technologies needed to transform society and culture and along with them infrastructures and institutions, in order to gain the full benefits of increasing exploitation. For the time being, most customers were simply not online due to the lack of infrastructure and the comparatively low level of the general intellect in the use of ICTs. Furthermore, neither netizens nor business models evolved enough to exert and exploit the transition from the customer-vendor relationship to the user-provider one.

Tellingly, the rising star of the dotcom boom was Amazon, whose initial business model treated Internet users as customers for paperbacks, and the Internet as an mail order system whose competitiveness lies in eminently efficient automation. It lacked the broad cultural mission and universal range of services that characterised Google, the company that epitomised the rise of social media monopolies in the next and current bubble. The former's success was based on its blatantly primitive industrialisation of the traditional service sector establishing continuity with existing practices, while the latter's success is founded on its sophisticated appeal to hacker culture which is mediated to the masses to partake of it. In simpler words, Amazon rose above its competitors during the dotcom boom because it was concerned with materiality but the competitiveness of Google is squarely concerned with information. Nota bene: according to the logic of disruptive innovation the examples of the few major successes in these two periods are only descriptive of their respective periods as far as we recognise that they went *against* the spirit of the times, in contrast to their myriad challengers.

situation). Indeed, political economic logic and its ideological legitimation changed more substantially between decades than between regions.

Pynchon (2013) evocatively captures the bleak atmosphere after the bubble burst and the stock market crashed: one could say that the firewall of the Internet separating an ideologically animated world from a materially driven one fell only a decade later than the Berlin Wall. (Except that the fall brought disappointment rather than enthusiasm.) As business models foundered, capital flee, developers lost their rock star status, and even the face of the Internet – e.g. the aesthetics of web design — assumed a more sombre and sober tone: blue and banal. I argue later on that the trauma incurred to both to the industry at large and to its foot soldiers in particular was a significant psychological factor in the further development of hacker culture in general and the rise of hackerspaces in particular.

Therefore, perhaps the most radical turn-around in hacker culture – as much as in the scholarship on ICTs and especially the Internet – has been the collapse of the "cyberspace" imaginary: the idea that "virtual reality" exists semi-independently or even against "meat space". Bruce Sterling wrote that

> Cyberspace is the "place" where a telephone conversation appears to occur. Not inside your actual phone, the plastic device on your desk. Not inside the other person's phone, in some other city. The place between the phones (1992, 11).

Even though it inspired some of the greatest feats and strongest experiences in the history of hackers, such ideology became untenable in face of the increasing repression and exploitation which resulted from the attention the Internet in general and hackers in particular received from the state and capital. In conjunction with the political economic shifts in power, wearable computers, locatory media, augmented reality, the Internet of Things and finally the physical computing that brought about hackerspaces has changed the perception of ICTs as a world beyond the screen (similar to how other mass media like photography, cinema or television were also experienced). Not that these could break the grip of cyberpunk on the hacker's imagination: while networks moved to the background, implants came into the foreground of fantasies. The corollary was that hackers did not play and fight in their own territory any more: hacking would have to liberate everyday life or transform it into a veritable battlefield now. Neither creativity nor conflicts can be constrained to cyberspace any more.

If cyberculture in the 1990s emphasised the discontinuity between virtual reality and the "meat world" in metaphysical terms, as the dichotomy collapsed, the 2010s were more about the continuity characterised by the connective terms collected above like "physical computing". The distinction is perhaps most characteristically captured in the contrast between the 1990s jargon IRL (In Real Life) and its contemporary rendering, AFK (Away From Keyboard). The docudrama *TPB AFK: The Pirate Bay Away From Keyboard* capitalises on this when the prosecutor uses the dated expression — that then have to be explained to the judge – and TPB founder Peter Sunde corrects him from the prisoner's box: "We don't like that expression. We say AFK - Away From Keyboard. We think that the internet is for real." 1999, between the two eras and just as the doctom bubble burst, Seattle saw one of the first massive mobilisation of alterglobalisation activists: the most prominent provider at the time (the radical technology collective Riseup) promoted its online services with the slogan "Get off the internet, I'll see you in the streets!"

In conclusion, the 1990s brought about three changes in the structure of the hacker scene. First, hackers raised their profile to the mainstream, capturing the public imagination. Tragically, hackers made the news mostly through repression – as a fascinating icon of transgression and subversion, as much an objects of desire as a desire to be repressed. Comically, the second coming of hackers in the 2010s meant a breakthrough to the mainstream as a much more hegemonic imaginary: even corporations and governments were eager to associate with the hacker moniker, even though at the same time acting against the "real hackers" with augmented zeal. The low political profile hacklabs kept during their existence can be explained by the former wave, and the high public profile – benevolent and apolitical – hackerspaces maintain can be explained by the latter. Second, the 1990s saw the establishment of "hacker circuits" – separate ones for North America,[56] Southern Europe[57] and North Europe[58]: in-person meetings where new members could be initiated, common knowledge and experience consensualised, central issues settled. As explained in other sections of this chapter, the circuits served as the site of emergence for both hacklabs and hackerspaces. Indeed, hackers often speak about such international encounters as temporary hackerspaces or proto-hackerspaces whose experiences were made permanent and developed through establishing local hackerspaces in their home towns. Third, the 1990s is the era of institutionalisation where an assortment of major and a host of minor foundations is established to facilitate the "legalisation" or "formalisation" of hacking. As bad as it sounds, these were set up largely independent of the organs of the state and capital, based on the relative autonomy[59] of the hacker scene. As a result, they arguably succeeded in providing organisational and financial sustainability as well as legal protection for the work of hackers without interfering in the practices more than necessary. Hackerspace foundations – some meta-foundations for helping the regional development of hackerspaces and ad-hoc foundations for the legal implementation of singular hackerspaces – were set up according to such a model around the turn of the 2010s when the approach has been tried and largely accepted.

[56] Like the abovementioned HOPE conferences.

[57] Explained in the first subsection of the Hacklabs section above.

[58] Explained in the next section, and briefly ibid.

[59] Defined later.

All three developments evolved in conjunction with the others, but involving different configurations of publics: without complicating the issue, it can be argued that institutions negotiated between the internal fora ("hacker circuits") and the mainstream public image of hackerdom.[60]

We can roughly discern three eras of hacking which preceded the era of hackerspaces: the golden era of hacking, the era of institutionalisation and the era of capital investment – if not historical periods, these can be seen as parallel processes which shaped the world of hacking throughout the years. In the general scheme of things we can also say that it took capital around a decade to notice the Internet itself as a major arena of capital accumulation, and the state at around two decades to recognise the same as a main locus of social control. As argued before, hackers by then treated it as their own territory and defended it vigorously, even though the radical imaginaries associated with it fell apart when it received serious attention to the state and capital, since hackers simply had no resources or ideological clarity to challenge these intrusions. Nowadays it is perhaps fair to say that where politics comes into play, the hacker scene is mostly fighting government control over citizens.

Note that it would be a mistake however to construct a simple dichotomy between the hacker scene and the state and capital – Weber (2004) for instance is right to state that the history of FLOSS is basically the history of corporations. There has always been a strong alliance between the hacker scene and capital, as well as at least the academia, if not certain organs of the state. Nowadays the most high profile technical and political organisations of hackers — Tor[61], Tactical Tech[62], EFF[63], and even a few hacker- and makerspaces – often receive support from the State Department of the US, most probably because they contribute to the achievement of policy goals such as delivering democracy to oppressive regimes.[64] The real question is how it is possible that the ethos of hacker culture has been preserved for so long in the middle of widespread collaboration and cooptation from left, right and centre. My answer is the theory of a relatively autonomous culture, expanded at length in a separate section further down when the physiognomy of hackerspaces has been established in detail.

6.3.1.2 Trajectory of hackerspaces As Figure 4 suggests, hackerspaces developed in three phases – roughly corresponding to the theory proposed by Farr

[60] Beyond the mainstream public image, a set of practices also solidified around this time, such as procedures for the responsible disclosure of software and systems bugs, patent regimes for software, laws regulating export restrictions on encryption, penetration testing, reverse-engineering, etc.

[61] Anonymity software development.

[62] Operational security education.

[63] Mostly legal counsel.

[64] For example in conjunction with the *Internet in a Suitcase* initiative announced by Hillary Clinton in her capacity as Secretary of State.

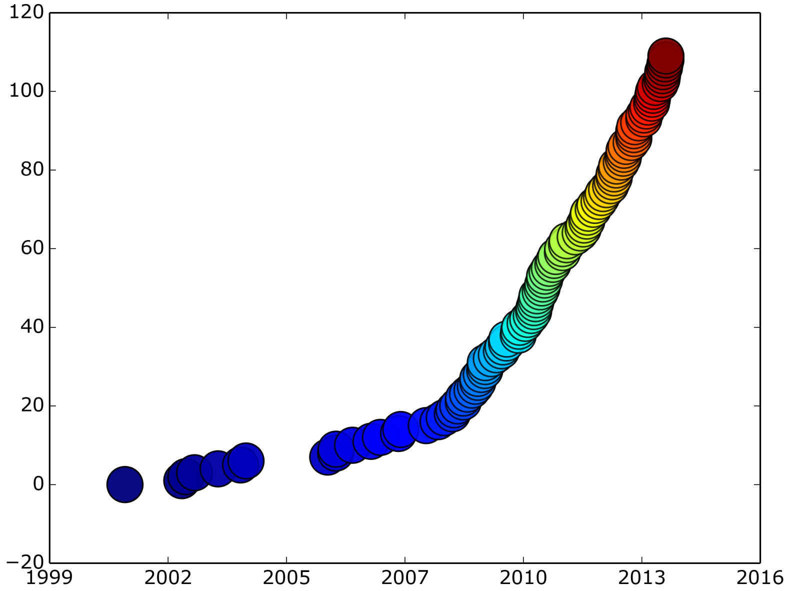


Figure 10: Registration dates of hackerspace domains from hackerspaces.org, based on the whois database. Own work.

(2009). First came the Ur-hackerspaces, which eventually inspired a few others in Germany, but only in the third phase can we speak about hackerspaces as a wide-spread and generally recognised genre in hacker culture at large. Hang-outs of early groups such as the Chaos Computer Club (founded in 1981 in Berlin) or the circle around Hack-Tic magazine (founded in 1989 in Amsterdam) have served similar purposes as hackerspaces for long, as well as their US counterparts.[65] All these however were quite exclusive, invite-only, at at least hard to find. Farr (2009) writes that "I wish New Hack was still around, just down Market street under the 'We Buy Diamonds' awning, only evidenced by a buzzer button labelled 'SETEC Astronomy"'.

However, the veritable Ur hackerspace — c-base in Berlin – was only founded in 1995. Presented as a space station buried under the German capital,[66] it was the first public institution to be established as a shared venue, meeting place and laboratory for hackers. It has actively addressed the public and involved various geeky publics in hacker culture. Metalab (founded in 2006 in Vienna) probably pioneered 24 hour access as a trademark feature of hackerspaces. It steadily grew into one of the most populous hackerspace in Europe, so that by now there is actual work going on every minute of the day. It is an interesting question how the hacker scene could find so many willing and able members to stock hackerspaces all around the continent in just a few years. My claim is that the previous phase of institutionalisation created a wide platform for enrolling new participants in the scene and socialising them in the normativity of hacker culture. Echoing Riemens' comments quoted earlier, Farr (2009) states that by this time the hacker scene was ready to open up to wider audiences, maintain its own public profile, and accept coexistence with official institutions: "Hackers could be perfectly open about their work, organise officially, gain recognition from the government and respect from the public by living and applying the Hacker ethic in their efforts." I repeatedly argued that in particular hacker meetings are instrumental for sourcing, bootstrapping and sustaining hackerspaces, and I will show in more detail some anecdotal evidence how that happens.

The best sources about the genesis of third wave hackerspaces is Bre and Astera (2008), Farr2009a, with a concise summary in Toupin (2014) – unfortunately, all written from a North American point of view. Thanks to much discussion around the topic, the origin myth is rather canonical by now. Nick Farr from the now defunct Hacker Foundation organised a collective trip called Hackers on a Plane in 2007 (Borland 2007) seeking to introduce local audiences to the budding *second wave* hackerspaces in Europe. Perhaps his efforts catalysed the moments of self-reflection to systematise and promote the model pioneered by German speaking hackerspaces like Metalab in Vienna, C4 in Cologne, and c-base in Berlin, amongst others. The trip led connected the DEF CON in Las Vegas, USA to the Chaos Communication Camp in Finowfurt, Germany, visiting some of

[65] Spaces like the L0pht, New Hack City (Boston and San Francisco), the Walnut Factory, the Hasty Pastry, and many other First Wave spaces that date back to the early 1990s are the stuff of legend.

[66] More on this in the second case study.

the key hackerspaces of the time in between. At Cologne in the C4 hackerspace Ohlig and Weiler gave the first version of what became the commandments of third generation spaces: the Hackerspace Design Patterns:

> In 2007, a number of meek and lonely hackers from the States went on the Hackers On A Plane adventure going to Chaos Communication Camp and then travelling around Europe visiting hackerspaces. When they arrived at C4 in Cologne, Jens Ohlig and Lars Weiler gave the first presentation of the Hackerspace Design patterns. It's a document made with the wisdom of doing it wrong in so many wonderful and disastrous ways (Bre and Astera 2008, 92).

The two went on to give a similar “Building a Hackerspace” (2007) talk at the 24th Chaos Communication Congress (24C3) to introduce the ideas to the scene as a whole. Perhaps the most decisive points were the following. Once, to argue that hacking is best done in collaboration at a shared social space. Twice, giving a systematic (or systematically looking) guide to making such spaces. Thrice, to actually oblige hackers to found hackerspaces. Bre and Astera (2008) captures the experience of listening to the preaching and realising that evangelising the hacker ethics is now a moral obligation. Page 53 reads thus: “They realized they lived in the same borough of New York City, and not only could they get themselves a hackerspace but they were morally obliged to do so.” Many hackerspaces in Europe and the United States started galvanised by these ideas that year. The next year Farr and friends organised a panel at the foremost North American hacker meeting (HOPE[67]) with the ambitious and obliging title “Building Hacker Spaces everywhere: Your Excuses are Invalid”. Thus, the growth in hackerspaces continued in all continents. Hackerspaces.org, an aggregator, directory and networking site was launched at the end of the year at the 25th Chaos Communication Congress (25C3). From the timeline it is evident that hacker meetings were instrumental for promoting the hackerspaces ideas: “Rodney [a founder of the Ductape hackerspace in Durban, South Africa] first heard about hacker spaces while listening to recorded talks from one of the HOPE conferences(Hackers On Planet Earth).” (Bre and Astera 2008, 73) Information is not necessarily the same as inspiration, however it seems that the campaign was moving enough to get things rolling: “they didn't have the cash on hand to really start a new hackerspace project, but after The Last HOPE, a hacker conference in NYC, we knew we had to do it” (Bre and Astera 2008, 65).

In Eastern Europe the foundation of the hackerspace in Budapest was also triggered by the 2008 talk, in combination with the positive experiences during visits to Metalab in Vienna nearby, with one person giving up their job in Germany to be able to work on it. Within a year the hackerspace opened with an epic party that passed on the spirit of the movement to hackers from the surrounding countries. New hackerspaces opened in Slovakia (Progressbar in

[67] Hackers On Planet Earth

Bratislava) and Czech Republic (Prague, Brmlab) in a few years' time. Second wave spaces already existed in the capital of Slovenia (Kyberpipa 2001, Ljubljana) and Croatia (mama hacklab 2004 or 2010, Zagreb). Soon hackerspaces in Poland were established too, so we can say that by 2010 major cities in Eastern Europe had hackerspaces.

As the figure above shows the growth of hackerspaces have not slowed down a bit since. While a few hackerspaces reportedly closed down, now it is not uncommon to find several active hackerspaces within a middle sized city on the Northern Hemisphere. Thus hackerspaces really ran viral and self-reproducing, on a much bigger scale than hacklabs before. Debates about institutionalisation and politicisation ensued. Here I take three debates which happened in recent years, following the expansion of the hackerspace movement.

First, as hackers started to share their workshops and their social spaces with each other, debates about sharing homes surfaced in multiple hackerspaces. Metalab in Vienna, H.A.C.K. in Budapest, Mama in Zagreb just as well as Noisebridge in San Francisco[68] and Sudo Room in Oakland had regular participants sleeping in the hackerspace. Hackerspaces often have showers installed and almost always equipped with a kitchen, so they are in effect viable spaces to use as a home: indeed, the very idea is that members should feel at home in their club. Moreover, following the long tradition of Gentlemen's Clubs as much as the more recent one of occupied social centre, some hackerspaces have places for visiting hackers to sleep for a few nights. However, neither have been envisioned as actual homes. The debates which I tracked are mostly over by now and went different ways. My impression is that spaces with over a hundred participants and actual twenty four hour opening times could not tolerate "squatters" in their midst, while less populous communities could appreciate the extra uptime and enhanced maintenance which resulted in people tacitly moving in.

In any case, these conflicts ensued because the social dynamics catalysed by the hackerspace model went beyond the model itself: the social architecture invited uses beyond those envisioned by its designers. As a response, the *hackbase* concept have been proposed and partly implemented in the Lanzarote, Canary Islands (2011) in a now defunct hackerspace called Cyber Hippie Totalism. The hackbase concept recognises that the positive experiences of hackers working and socialising in a common space invite sharing a complete life with each other. The concept was further discussed and developed in a meeting on cooperativism organised by Hackerspace Brussels in 2014, with the idea of organising a large space for several projects which would have included living quarters. Sudo Room initiated another failed experiment along almost exactly the same lines: renting an post-industrial space of several thousand square meters for various projects including the hackerspace, a living cooperative and other friendly projects. The practice of course have existed since the dawn of hackerdom. It is interesting to remember that Levy (1984) reports hackering at the old MIT AI lab removing ceiling tiles or hiding mattresses so that they can sleep in their offices. Many

[68] Located in the Mission district sometimes associated with homelessness.

private hacker homes include spacious working and social spaces as well as guest rooms in the house, but these are invite-only reminiscent of second wave hackerspaces. Of course, since hacklabs were located in occupied social centres that often had people living there full time too, this has always been part of their dynamics. Even in hacklabs like Riereta where nobody officially lived, it was uncontroversial to “camp there”, sleep in the back room and use the communal shower, etc. In sum, at least some sectors of the hackerspace ecosystem nurture a desire for a shared life, and the coming years may bring some sustainable experiments derived from the hackerspace model which move in that direction.

Second, an attempt to take over the makerspaces brand by Make Media using military funding is documented by Altman (2012a). Maker Media – a kind of physical computing equivalent of WIRED – is publishing MAKE magazine and organising the Maker Faires, both of which provide forums for DIY/DIWO (Do It Yourself / Do It With Others) type of tinkering. The magazine is often stocked in hackerspaces and hackerspaces sometimes participate in Maker Faires, primarily in the US. WIRED and Maker Media are often quoted as prime examples of the Californian Ideology and its discourse in action. Mitch Altman – like Nick Farr mentioned above – is a key figure of the hackerspace movement who spends his time travelling the world, visiting hackerspaces, developing open hardware and mainly mentoring hackers. His soldering workshops are a staple of hackerspaces and hacker meetings and his inventions like the Brain Machine is widely acclaimed in the scene. For instance the latter is included in the logo of Hackerspace Brussels, depicting the founder of cybernetics Norbert Wiener wearing one.[69] Altman was awarded the first Maker Hero by Maker Media.

While hackerspaces and makerspaces are often not distinguishable through how they look like and what is actually going on in the spaces on a daily basis, their discourses increasingly form two separable streams. There were three major factors which contributed to the discursive shift towards the makerspace concept in parts of the hackerspaces milieu. Firstly, with the advent of physical computing and the emerging focus of hackers on the development of tangible artefacts, some participants felt that the traditional hacker domains have been surpassed. Secondly, as hackerspaces gained popularity and started to address a wider audience, many felt that the “hacker” moniker which worked well to attract young and middle aged white males mostly with an interest in computing was often a hindrance when trying to integrate people who did not match that profile. Thirdly, and particularly in the United States, the influence of Maker Media which published the seminal organ of the new movement (the aforementioned MAKE Magazine) was felt.

MAKE founder Tim O’Reilly has been a long term advocate of FLOSS, and in fact he has been one of the persons behind a similar discursive shift: from *free software* to *Open Source*. **The idea behind Open Source Software was to make free software more attractive to the industry through**

[69] Interestingly, the brain machine itself is based on the invention of Gyson, another cybernetics pioneer.

playing down or neutralising the political/ethical values which many saw as embedded in the code. The transition from hackers to makers had a similar rationale, creating a more politically neutral, socially welcoming and professionally credible image for community workshops. A good example of such distinction in actual discourse is when a maker who recently adopted the identity gets offended for being called a hacker, explaining that makers are different from hackers: "because hackers never finish things properly"[70], that is, they are not reliable and productive. That said, (in contrast with the open source vs. free software issue) the discursive shift or rift is still far from completion. For instance the Wikipedia entry for Makerspace – for the moment – redirects to Hackerspace which reflects quite well the current state of representational processes in the net savvy hacker/maker milieu.

The controversy broke out around the announcement by MAKE Magazine that they received 10 million dollars for bringing maker tools and maker culture to 1000 high schools in America and around the world (Dougherty 2012). By this time many makerspaces and hackerspaces were focusing on educating teenagers, yet many saw it as a problem that the funding for MAKE came from the Department of Defence through the DARPA (the Defence Advanced Research Projects Agency). Questions have been raised about the idea of hackers implementing military programs for teenagers. Furthermore, the initiative which came out of this grant revolved around a new website, makerspace.com, and branded as the Makerspace program. The website would gather a database of makerspaces compatible with the program, and provide advise to participating organisations from schools and makerspaces. Some saw this as a replication of effort, since hackerspaces.org existed since 2006 as a community effort which famously catalogued more than 1000 hackerspaces, and aggregated much documentation on the subject. While the makerspace.com website would take over part of the discourse around hackerspaces, it would also overshadow the discursive space where makerspaces which are not affiliated with the program could communicate independently.

The grant was received by O'Reilly's MAKE division in partnership with Otherlab. The two persons behind the grant application were Dale Dougherty, co-founder and editor in chief of MAKE Magazine, and Otherlab's Saul Griffith, a scholar and entrepreneur who is also a columnist in the magazine. They are both deeply embedded in the scene and earned much respect through their work. The goals of the program – as far as I could make out – are threefold. Firstly, (1) to develop low-cost options for making makerspaces in educational institutions. Secondly, (2) to develop an online collaboration and documentation platform for educators, children and interested makers/hackers. Thirdly, (3) to use Maker Faires – gatherings of makerspaces which has many regional versions in the United States – for promoting and showcasing the results of children's' projects. All in all, it seems to be about bringing makers and educators, as well as makerspaces and schools, even closer together. Interestingly most makerspace participants more

[70] Personal communication with Debora Lanzeni, 2013 August.

or less agree with the above goals, or at least do not see them as controversial in themselves.

Compared to the discourse about openness and collaboration which prevails amongst makerspaces and even Make itself, there are few public details about the actual conditions and implementation details of such DARPA funding available on the Internet. Neither the official Makerspace website nor the “Playbook” guide developed for the programme provide much information about it. This could have been a factor in the growing controversy, since given the silence from the applicant’s side, a lot of the contextual information comes from DARPA itself. The grant is part of DAPRA’s MENTOR program, which stands for Manufacturing Experimentation and Outreach, itself only a part in the AVH (Adaptive Vehicle Make) portfolio. DARPA defines them in the following:

> The Defense Advanced Research Projects Agency (DARPA) has embarked on a series of programs aimed at revolutionizing the way defense systems and vehicles are made. Titled Adaptive Vehicle Make, the portfolio has three principal objectives: to dramatically compress development times for complex defense systems such as military air and ground vehicles, to shift the product value chain for such systems toward high-value-added design activities, and to democratize the innovation process. The Manufacturing Experimentation and Outreach (MENTOR) effort is part of the Adaptive Vehicle Make program portfolio and is aimed at engaging high school students in a series of collaborative distributed manufacturing and design experiments. The overarching objective of MENTOR is to develop and motivate a next generation cadre of system designers and manufacturing innovators, and to ensure that high school-age youths are exposed to the principles of modern prize-based design and foundry-style digital manufacturing (DARPA 2010).

It is not difficult to read the keywords associated with the makerspace movement between the lines of this announcement written in the language of the military-industrial complex: rapid prototyping, DIY (Do It Yourself) culture, personal manufacturing, etc. Therefore, it can be argued that *the call was effectively geared towards makerspaces*, or at least inspired by the innovation they engendered in the recent years. Advocates - and many of them! – saw it as the long-awaited sign of recognition of their relevance from the mainstream world, and an opportunity to trespass their niche and make a wider impact on society at large. Other hackers and makers suspected cooptation.

The *critical discourse was kick-started by Mitch Altman*, who used his position as a MAKE writer and sometimes Maker Fair organiser, and above a Maker Hero, to thematise the problem. As he summarised in an invited comment in the Journal of Peer Production (Altman 2012b), all his previous employers have been approached by the military, and he subsequently left in protest. In line with this policy he announced that he will not be involved with MAKE Magazine and Maker Faires in the future, even though he still thinks they are good initiatives overall. Instead, he organised a panel at HOPE (the biannual hacker conference) on the issue of DARPA funding.

At the panel, the most erudite comments came from Fiacre O'Duinn (a person known to build bridges between librarians and makers, working on the possibility of turning libraries to makerspaces) who provided some background analysis about the recruitment policy of the US military and pointed out that high schools are a seminal site. He framed the issue as the military's attempt to reach small kids, and referred to a study by ACLU (the American Civil Liberties Union) which recommended against recruiting efforts targeting children below 17.

On the rougher side, it was reported (letter to the noisebridge-discuss on May 23, 2012 by Corey McGuire and editors (2012)) that **Jake Spaz from the Noisebridge hackerspace walked around with a protest sign** including the image of a paedophile who is showing his genital organ to a child accompanied by an asserting young man. The latter was identified as the likeness of Dale Dougherty and the former pictured wearing a coat with the DARPA logo on it. The child was not identified. The caption read "Hey, Maker Faire! Don't expose children to DARPA and the military!".

Even though the whole project was a done deal when it came to light, according to my informants, the implementation of the program met tacit resistance from hackerspace/makerspace communities and is practically thwarted by the difficulties of collaboration between organisations with top-down and bottom-up managerial cultures. A similar controversy about the infiltration of the Dutch hacker camps by the national cyber police in conjunction with local cyber-security companies connected to the hackerspaces scene is detailed in Maxigas (2014b). Smaller examples could be recounted from several other European hackerspaces, but the one here is perhaps the most descriptive of the tendency of state and capital to include hackerspaces in their recruitment and educational as well as research and development frameworks. As stated before, such trajectory of cooptation rests on the genuine, recent tendency of hackerspaces to turn to younger audiences.[71]

Third, reminiscent of discussion of the *activist ghetto* in hacklabs, discussion on the limits of openness as it is practiced in hackerspaces came to the fore in recent years. The focus of conversations and conflicts were feminist critiques of male hegemony in the hacker scene in general and the hackerspaces in particular. As the hacker idea moved to the mainstream, discussions about inclusivity became more common, and feminist hackers could regroup and become a vocal minority in the scene.

[71] Two historical antecedents are worth mentioning: U23 nights at the second generation hackerspace C4 which allow only people below the age of 23 to enter the space, (Bre and Astera 2008, 11) and the R.E.S.I.S.T.O.R.S. barn of the 1970s which was a children's computer club (R.E.S.I.S.T.O.R.S. 2009).

The formation of an organised feminist block closely follows the pattern of hacker history outlined above, proving that feminist streams have been part and parcel of hacker culture from the beginning. The prehistory of hackerdom and the cabals of the first generations often included women, who were actually in majority in many areas of computing before it became a lucrative and respected profession. The platform for feminist activities was built in three steps. (Toupin 2014) Initially the Geek Feminism Wiki allowed hackers to find each other and exchange information, reminiscent of the BBSs of early times. As women started to talk to each other, analyse their experiences and form a discourse, it became clear that harassment of women in the hacker scene (too) was a rampant problem. Secondly, following the step of institutionalisation, the Ada Initiative was set up in 2011 to "support women in open technology and culture through activities such as producing codes of conduct and anti-harassment policies, advocating for gender diversity, teaching ally skills, and hosting conferences for women in open tech/culture."[72] The AdaCamps organised by the same foundation from 2012 became the most prominent meeting places of feminist hackers, structurally similar to the hacker meetings established in the second half of the 1990s partly in response to repression. Thirdly, in recent years the specific conversations and conflicts in the hackerspaces led some hackers to establish their own, gender-aware and gender-oriented hackerspaces. The idea of establishing such spaces spread as fast as the idea of vanilla hacklabs or hackerspaces. Toupin (2014) reports that

> the following women-centered/feminist and/or people of color-led hackerspaces have emerged in the past years: Mz Baltazar's Laboratory in Vienna (feminist space created in 2008-2009), Liberating Ourselves Locally in Oakland (people of color space created in 2012), Mothership Hackermoms in Berkley (women-centered space created in 2012), Seattle Attic in Seattle (intersectional feminist space created in 2013), Flux in Portland (intersectional feminist space created in 2013), Double Union in San Francisco (intersectional feminist space created in 2013) and Hacker Gals in Michigan (women-centered space created in 2014).

That is seven spaces founded in the course of a few years and it seems that more are in the works (for instance FemHack in Montreal, Canada). The significance of separatism should not be underestimated. On the one hand, it is an process which changes the strategic situation in all hackerspaces, not only gendered ones. Feminist advocates earn reputation and legitimacy by moving beyond pure critique to constructive and practical steps which lead to technical contributions and community organising — work whose import even their harshest critiques cannot deny. The stronger position of feminist hacking, and the threat of loosing members to gender-aware hackerspaces forces all hackerspaces to take their concerns more seriously. On the other hand, the feminist critique of openness is a political critique of openness in particular and hegemonic liberal/libertarian

[72] https://adainitiative.org/

discourses in general. In case it takes root in the hackerspaces it opens new perspectives for the politicisation of discourse about technology, the role of engineers in society, and the socio-political import of hackerdom. This is already apparent in the intersectional orientation of some feminist critique: according to Toupin, intersectional feminism takes into account that male hegemony is but one dimension of oppression along with race, class, age and countless other lines. Of course the limitation of such critique is apparent too in that it is easily turned into a liberal centrist argument about the enrolment of all subjects in social production rather than in the contestation of liberalism, capitalism and ultimately, modernity. At the same time there is amble space for advocating the adoption of an anticapitalist stance within the intersectional discourse too, and probably both tendencies will exert their influence. For now what can be stated with confidence is that in recent years some hackerspaces included political concerns in their public profile and official mission, an event which can be understood as a turning point in the history of the hackerspaces model reminiscent of hacklabs.[73]

To recapitulate: the three contemporary controversies explored here were the following. (1) Sleeping in hackerspaces, which points to the establishment of live-in hackbases; (2) cooptation into the military-industrial complex, which points to the relevance of hackerspaces to the ambitions of the state and capital; and (3) holding women only nights in hackerspaces, which points to the critique of openness and the political neutrality inherent in the original hackerspace model. Interestingly, all three of these controversies bring back concerns that were simply part of the environment for hacklabs.

Parallel to these debates, a whole genre of shared machine shops which further institutionalised and commercialised the hacklab model sprang up. Fab Labs, – initiated by the MIT Laboratory for Bits and Atoms – have been established all over the world, often tied to academic institutions, with an increased focus on distributed manufacturing and rapid prototyping. Tech Shops are a chain of stores selling access to the typical equipment of hackerspaces, like 3D printers, CNC machines and laser cutters. Men's Sheds started in Australia with the idea of establishing social spaces for elderly males, but have broadened and diversified their missions since then. These three actors – the academia, the industry and the state – look for ways to match the model of grassroots communities to their respective aims of enhancing formal education, reinventing business models and delivering welfare services. A more detailed but still non-comprehensive tally of such genres is included in the last section of this chapter.

Given the high speed of the recuperation of the hackerspaces model and the explosion of genres in the shared machine shop field, hackerspaces had to differentiate themselves from other kinds of shared machine shops – and they primarily did so by emphasising their value as informal organisations managed by their

[73] Note that while feminism does not constitute such a totally comprehensive life world like autonomous politics, it does have an analysis of society as a totality which can serve as a coherent ground for critique: exactly what hackerspaces were missing.

participants, presenting themselves as self-organised engineering enthusiasts. The external pressure was no doubt a significant factor in creating coherence and stabilising the hackerspace model. In recent years people who are interested in more commercial spaces can often find one (for instance a Fab Lab) in their home town. On the one hand, the semi-autonomy of hacker culture in general combined with the already established circuits of hackerdom in particular somewhat sheltered hackerspaces from recuperative processes. The institutionalisation of the previous decade which was in part a response to repression helped hackers to pull together and defend their values and practices. Overall, in the 1990s hackers found their greatest enemy and (particularly in Europe)[74] greatest sponsors in the state, by the 2010s it was capital that often challenged and sometimes supported hackerspaces.[75] In summary, recent years showed an increased reflection on the role that hackerspaces in particular and engineering in general play in the political construction of society. On the one hand these reflections brought back many discrete elements of the debates customary in hacklabs such as the division of sharing a living space, collaboration with the state and capital, and questions of openness particularly about feminism. On the other hand these reflections did not happen from the more clear and solid ground of a wider social movement as in the case of hacklabs. However, a possibility for another way of socio-political intervention in political processes rooted in an alternative engineering culture seem to have opened up.

6.3.1.3 Hackerspace participation at hacker camps The biannual rhythm of hacker camps in Europe is established by tradition so that Dutch hackers organise a camp with a different name each four years since 1989 (the Galactic Hacker Party), complemented by the Chaos Communication Camp organised by the Chaos Computer Club (a German speaking hacker organisation) every other four years since 1999. While the latter events are in the hands of a stable organisation (even if it has “Chaos” in its name), the responsibility to organise the Dutch camps has been passed around between various groups and ad-hoc organisations. The camps were started by the notorious hacker crew Hippies from Hell, infamous for their exploits, the publication of the hacker magazine Hack-tic and for their mind blowing parties.

Since the last but one camp, Hacking At Random (HAR2009), a large number of hackerspaces have been founded in the Netherlands – as previously argued, partly thanks to inspiration and initiative drawn from the camp itself. As the website of OHM2014 states, “HAR2009 … has without a doubt been the epicentre from which a tsunami of hackerspaces spread out over The Netherlands.” Therefore hackerspaces and members took serious responsibilities in organising the next camp in 2013 entitled Observe! Hack! Make! (OHM2013). In fact the legal organisation which was set up to coordinate the event (the IFCAT Foundation)

[74] Metalab and Mama for instance received much state subsidy for their functioning.

[75] H.A.C.K. in Budapest often receives donations from tech corporations, which contribute to its sustainability.

held its first board meeting at the Sk1llz hackerspace in Almere, while the kickoff party took place at the Hack42 hackerspace in Arnhem. Two of five board members were founders of other hackerspaces, in Den Haag and Amersfoort. The organisational team explicitly called on hackerspaces to get involved, viewing them as the offspring of the last conference and the main stakeholders of the next one: "Time to close the circle: hackerspaces of the nation, join your forces to create a place-time of wonder!", exhorted the organisers.

I made field visits to 7 out of the 11 active hackerspaces in the Netherlands the December before OHM took place. Members of all these spaces except one planned to attend OHM. For instance, the relatively small BitLair contributed core members to both the lighting team and the Network Operation Centre. Similarly, at the event itself I could see the "branded" tents of most significant hackerspaces I know about in Europe, including those from my Germanic and Eastern-European field works. The dedicated wiki page of ACKspace (Heerlen, Netherlands) stated that "Like many other hackerspaces, some of us (ACKspace) are also going to attend this conference. It would be awesome to attend the conference as ACKspace hackerspace village." Interviews with participants also made it clear that the hackerspaces and their members have been instrumental and essential in making the event happen. Therefore it is not far fetched to state that the gathering was not simply a gathering of hackers from various scenes but the most significant gathering of hackerspaces in Europe. Other shared machine shops were also represented but not as prominently as hackerspaces. LAG hacklab members came individually, as well as members of some other hacklabs. One of the two Fab Labs in Amsterdam came with their fab truck, while the other (embedded in the Waag Society) did not have an official presence. This supports the hypothesis that shared machine shops form loosely overlapping scenes, as well as the one that the hacker camps are a venue primarily attended by hackerspaces.

Following the aforementioned hacker camp HAR2009, the first or second hackerspace to be established in the Netherlands was RevSpace (Den Haag), whose members have done much to promote the concept, including using the HXX Foundation (the legal entity behind the camp) to promote the hackerspaces model. For instance in subsequent spaces I learned that the bylaws for the foundation which provided the institutional basis were mostly taken from the RevSpace documents. The following years saw hackerspaces mushrooming around the country, with at least 9 other established by the time of the next hacker camp (OHM2013): ACKspace (Heerlen), Bitlair (Amersfoort), Frack (Leeuwarden), Hack42 (Arnhem), RandomData (Utrecht), Sk1llz (Almere), TechInc (Amsterdam), TkkrLab (Enschede), VoidWarranties (Antwerp).

6.3.2 Conditions of emergence

Even though there is a significant amount of pure software development going on in these spaces, as well as many other genres of hacking, what really brought people together is hardware hacking. But how hardware hacking emerged from its historical obscurity? The ur hacker Voja Antonić, who single-handedly built the first personal computer in Yugoslavia said that the first hackers were necessarily hardware hackers, because first they had to build a computer or similar machine for themselves [-Antonic2014a]. Indeed, the following decades saw a comparative decline in the popularity of hardware hacking, DIY electronics construction and open hardware design. This tendency changed in the 2010s and this section proposes some explanations as to why it happened so. It is safe to assume that a variety of causes of different nature played a part in the historical shift.

(1.) Similarly to the 1970s when microcontrollers appeared on the mass component market, by the 2010s programmable microcontrollers became widely available on the consumer market. Programmable microcontrollers are very cheap chips which are traditionally programmed in machine language (called Assembly) or in low level languages such as C, using quite complex programming boards. Once a chip is programmed, one can build it into a device to perform logical operations between inputs and outputs: for instance, it can turn on a LED when a movement sensor is activated. Besides the chips, tens of thousands of other basic components like LEDs, sensors, relays, and resistors are available in electronic shops for cents.

(2.) One of the most successful open hardware designs of all time, the Arduino board made them very easy to use for rapid prototyping. Arduinos make the whole process so easy that microcontroller programming is brought from the realm of engineers to the realm of the masses, who can now produce original designs in a couple of hours even if they have not worked with hardware before. The Arduino is programmed through a user friendly interface (Integrated Development Environment) run on an ordinary PC connected to the board through a conventional USB cable. The board includes pinholes connecting to inputs and outputs, like the led and the movement sensor in the previous example.

(3.) Since the burst of the dotcom bubble after the turn of century, there was a general lack of enthusiasm in the tech community towards building more websites. The rising stars of the industry often had innovative business models but just as often they were technically boring. The early versions of Facebook, Twitter and Amazon are prime examples: a profile directory, a message syndicator, and a web shop. These are routine tasks for a webmaster to build even if they evolved into intricate contraptions distributed on an extreme scale since their foundation. The hegemony of certain technologies like the PHP programming language and the MySQL relational database server also turned away many hackers, because they found them technically stagnating and in any case fundamentally flawed in their design. At the same time the rising media monopolies meant that the basic actors of the market cemented their position, so that disruption became more difficult and more costly year by year. In practice, capital discovered, colonised and therefore stabilised the Internet as a sector of the technology market. In terms of affective history the 1980s and early 1990s felt like the Wild West[76],

[76] see the EFF rhetoric dissected above

the late 1990s and 2000s like a Gold Rush, and by the 2010s the railroad arrived and civilisation reigned supreme. It was increasingly unlikely that "something would happen" on the Internet for some time. After all the hype about Internet technologies, there was a period of attention fatigue and the creativity of hackers need another outlet.

(4.) Tom Igoe and Dan O'Sullivan published a book called "Physical Computing: Sensing and Controlling the Physical World with Computers." (Igoe and O'Sullivan 2004) The concepts set out therein — the idea that you can program, control, and communicate with things outside the computer – created a host of paradigms and practices that showed a meaningful and largely unexplored direction beyond the development of the computer. Similarly to the Internet earlier on, physical computing offered an unregulated environment governed by powerful collective imaginaries. Indeed, if cyberpunk was a thing of the 1990s, its promises have never been fulfilled. Adherents complained that instead of flying cars, wearable computers and robot slaves, the 2010s was buzzing with the ability to transmit 140 words messages without a wires. Therefore it was necessary for them to take future engineering into their own hands and realise the sci-fi visions which the market failed to deliver. Given enough DIY spirit, physical computing as a kind of *robotics for the masses* allowed many of these childhood dreams to became an everyday reality.

(5.) The alterglobalisation cycle of struggles largely subsided by 2007, leaving behind a painful vacuum in the mood of the political underground comparable to the disappointment of the dotcom boom in the tech industry. Since a global movement against globalised capital – with counter-summits where the opposition can concentrate its powers – was not viable any more, many activist retreated into the local scene and focused on creating communities according to some or other principles of the previous cycle. The conclusion set in that the alterglobalisation cycle closely mirrored neoliberalism in its political mechanisms For instance temporally the summit hopping was much like capital flight while the "global networks of resistance" were organised geographically from London, New York and Berlin, comprised by satellite groups in capital cities, and the countryside where still most people lived was virtually invisible on the planning table. After the alterglobalisation movement nurturing local contexts came to the fore but could never supersede the desire for a global movement. Even though hackerspaces were not affiliated with any of these, many early hackerspace organisers came from such background. Indeed, hackerspaces fulfilled both requirements: local and global at the same time, they could be meaningful, sustainable and innovative human environments yet fit together into a global movement of sorts. Additionally, many techies – such as myself – who were previously involved in media activism like Indymedia retreated back to more specific and specialised technical interests when the alterglobalisation movement lost its political impetus and inspiration. The pattern is obviously reminiscent of the relationship between the autonomous movement and the hacklabs which have been explored at length in the first sections of this chapter.

These contributing factors opened the window of opportunity where hackerspaces could possibly succeed as an organisational model. Hardware hacking necessitates the pooling of tangible resources: both in terms of components and lab equipment, as well as tacit knowledge and know-how. Of course open source hardware, electronics and robotics is most often coupled with logical control and processing implemented in software code, so that hardware hacking is not really the antidote of software development, but only a half step away from it. The increasingly popular knowledge of network programming and the widely deployed Internet infrastructure were precursors to such developments.

What happened next was also similar to the early days of hardware hacking. The subsequent range of technologies, including 3D printers, laser cutters, CNC machines (all digital fabrication tools), and quadrocopters (the hacker version of drones), DNA synthesizers, software-defined radios – were all built on the extended knowledge and availability of microcontrollers. What was built in hackerspaces found their way to the market in a few years through startup companies, few of which became leaders in the newly created markets.

7 Synchronic view: Social dimensions of hackerspaces

7.1 Hackerspaces as a black box

It is not far-fetched to argue that every few years hackerspaces absorb a major technology from the military-industrial complex,(The term is the original form of the "military-industrial complex" that Eisenhower warned about in his farewell speech, see Giroux 2007) and come up with a DIY-punk version to be reintegrated into postindustrial capitalism. Looking solely at the input and output of hackerspaces can be illuminating as to the question of what sort of dispositif is it: what are the translations, mediations and shifts that hacking performs on technologies? Reverse engineers often try to understand obscured systems simply by comparing the systematic differences between the output from various inputs. Here I take three examples of technological artefacts which were transformed by hackerspaces: 3D printers, drones and synthetic biology.

Basic 3D printers make tangible objects from digital models by melting and extruding plastic layer-by-layer. They are mostly used to print simple household objects, replacement parts or very customised small-series objects – and some 3D printer projects like the RepRap aim to produce self-reproducing machines that can print themselves. Since the technology is continually evolving, it became noteworthy even for mainstream news outlets to publish articles with headlines of the general syntactic form "3D printed X made by Y", for instance "3D printed violin made by University of Exeter". (University of Exeter 2012; CNN 2012) A practitioner explains what 3D printers do in these words:

> The most popular form of 3D printers available is FDM (Fused Deposition Modeling) which works by melting a plastic filament that is fed through a heated nozzle and layering it on a variable height platform all controlled by a software that divides a 3D object into many thin 2D cross-sections so it can print layer by layer into a 3D part. It is basically like taking a hot glue gun and layering the glue on a sheet of paper to make something 3-dimensional (Wires 2014).

As the foremost media organ of the hackerspaces scene (Hackaday) reports, fused deposition modelling (FDM) was invented and patented by S. Scott Crump in 1989 who founded the company Stratasys to capitalise on the idea. (Benchoff 2013) Stratasys had its initial public offering in 1994 on NASDAQ and still holds its position as the leader in the 3D printing market, despite fierce competition from established technology manufacturers like HP and the plethora of startup companies which emerged from the hackerspaces scene (MakerBot, Aleph Objects, etc.). The RepRap project was started in 2005-2006 at the University of Bath to create and open source self-replicating 3D printer. Shortly after the founding of the NYC Resistor hackerspace, members joined the RepRap community efforts in

2008 making major contributions to the development of the design. The second version of the printer was published the next year – the same year that the Stratasys patent by Crump expired: 2009. By this time NYC Resistor derived much of its public image and in-scene reputation from participating in the 3D printing world, like giving board members to the new RepRap Foundation.

The same year the patent expired, three DIY Resistor members created a spin-off company called MakerBot Industries. MakerBot Industries had the support of the community at first for its contributions to the development of the RepRap open source 3D printer, but when the company started to market its own printers with proprietary parts and solutions, the community reacted badly. It was the nightmare of any peer production project developing an open technology: the spin-off company capitalised on the freely sharable work of the community while not contributing back upstream (e.g. to the original project) any more. In practice, they were using the free innovation of the Really Free Market to bootstrap their closed source business model. Then in 2013 Stratasys acquired MakerBot. By 2014 Stratasys was suing the competitors of MakerBot – Small and Medium-sized Enterprises – for the violations of patents it filed in the meantime. With this the storyline came full circle.

Similarly to the story of the personal computer in the 1970s, hackers picked up a technology ripe in industry and academia to produce a simple and effective version of it which is easy to understand and use for personal purposes. The difference is that the trajectory was drawn in a much more institutionalised scene where stable organisations supported the research and development effort and stabilised legal practices (in this case open hardware licences) protected the results. Even so, just like with the personal computer, startup companies were formed in the next cycle with the mission to make the innovation available for the masses, and most of these companies failed miserably. The ones which emerged were acquired by already established players that started to kill off competition aggressively in order to acquire a quasi-monopoly position on the newly created market.

The transformation of drones to quadcopters is another example where we can observe the hackerspaces scene as a black box and trace what happens after the scene digested their next victim from the repertoire of the military-industrial-academic complex. Figure 5 shows Google search frequency for “drone” versus “quadcopter”. Capital letters on the timeline are noteworthy news articles, associating drones with the Unmanned Aerial Vehicles used mainly by the US military. All report drones killing people or people taking down drones, whereas the articles about “quadcopters” present them as technical curiosities used by common people. The chart shows that drones received mainstream exposure around 2010-2011, while the popularity of quadcopters picked up around slowly until 2013. Then on the trend lines synchronise, with the less significant quadcopters following the spikes in popularity of the more popular drones.

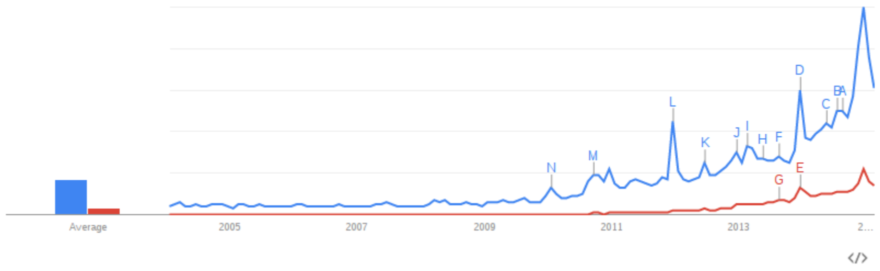


Figure 11: Google Trends: search keyword frequency for "drone" (blue) versus "quadcopter" (red) source

Unmanned Aerial Vehicles (abbreviated UAVs) or drones are basically small remote control helicopters pioneered in low intensity warfare situations by the US Army, which became useful because of their low risk operation, good cargo capacity for weapons and relatively extended range. They are obviously suited for difficult terrains and high precision strikes in situations where the loss of human soldiers would be undesirable. As Francis Fukuyama states, "This is a very seductive path to follow because it is relatively cheap, lacking the huge logistics trains that accompany conventional force deployments, and seems for the time being to be a monopoly of the United States." Quadcopters are the staple of hackerspaces and hacker meetings since 2013 or so. The idea is simple: put four stepper motors on a cross; control them with an Arduino on top; power them with a battery on the bottom. Add a wifi module and you have remote control. More powerful quadcopters can carry cameras which makes them much more fun. The design brought prices of DIY drones down drastically in the last few years, so that minified versions can be ordered from China for a few dozen Euros, and anything upwards of that. It also diversified the use cases for drones dramatically, so that besides military applications, spin-off companies started to introduce drones to fields as diverse as public policing, professional photography, parcel delivery, disaster response and agriculture.

Technically, the idea builds on the previous waves of technologies absorbed into the hackerspace milieu. The control unit more often than not built around an Arduino or similar rapid prototyping board, while the mechanical design is strongly reminiscent of the 3D printers described above: the engineering problem is to control four stepper motors in concert to achieve three degrees of freedom. Therefore we can observe a certain path dependency in the DIY works coming out of the hackerspaces: previous favourites feature in the next designs as members apply their accumulated knowledge and tools to new problems.

In 2007 Chris Anderson – WIRED magazine editor-in-chief and author of *Makers: The New Industrial Revolution* – founded DIY Drones, an open hardware and FLOSS effort organised as a peer community to build autopilot systems for radio controlled vehicles such as quadcopters. A recent post on the website details the stance of the Federal Aviation Administration's stance on model aircrafts, including drones: "DO fly a model aircraft for personal enjoyment – DON'T fly model aircraft for payment or commercial purposes." (DuCray 2014) In 2009 Anderson founded 3D Robotics as "the manufacturing arm of DIY Drones" (CrunchBase 2014) to sell the platforms which came out of the community. The most lucrative use case seems to be the agricultural mapping drones sold to Monsanto which help to place pesticides and report crop yields. (Raskin 2013) While selling to Monsanto, DIY Robotics also serves the hobbyist scene: for instance the famous political scientist Francis Fukuyama reports building their kits. (Fukuyama 2012) Currently the industry is waiting for the Federal Aviation Administration (mentioned above) to approve in 2015 the regulations for commercial drone flights which will allow Amazon for instance to fly parcels directly to US homes through its Amazon Prime Air program. (Administration 2015) Drones are already used on both sides of the Atlantic by authorities to "allow law enforcement agencies to intervene in the event of persistent disturbances that that move between areas — for example, a riot", according to the legislation approved in 2013. (Gijzemijter 2014; Kamer 2013)

Then in 2012 "the Pentagon reached out to open-sourcers through UAVForge, a project of the Defense Advanced Research Projects Agency", as a practitioner put it, "to accelerate the development of their drone technology," (Raskin 2013) Interestingly, none of the 140 teams managed to complete the baseline task of sending a live video feed without line of sight, even though other criteria – like cost below 10.000 USD – was easily fulfilled. (Drummond 2012; Warwick 2012) Thus this story also harked back to the origins it came from: the military-industrial-academic complex. UAVs were picked up by the hackerspaces, "domesticated" into the peer production environment, the hacker scene and in the technological repertoire of DIY enthusiasts. The activities therein brought down prices and found use cases for commercial applications that spin-off and startup companies tried to capitalise on. Finally, major corporations like Amazon as well as the military research and development agency (DARPA) took note of these advancements and built it into their business model and research agendas.

DIY biology is the new kid on the block around hackerspaces.[77] It is based on inventing cheap equipment, user friendly documentation, and finding opportunities for more or less lay people to participate in biological experiments or even research. DIY bio participants aim to replicate research by the academia and the industry, and contribute to scientific knowledge and its applications in some areas, but using laboratories and personnel in extra-institutional contexts. One condition of emergence of this sphere of activities is the Human Genome

[77] Thanks for Rosen Bogdanov Ivanov for providing background information for this part of the dissertation. Personal communication, 2015-02-18.

Project (1984-2003) proposed and funded by the US Government[78], implemented in a large network of universities and research centres. The HGP produced a database a full human reference DNA sequence, patched together from various samples. While work continues on the HGP, the next step is the Personal Genome Project (2005) that provides DNA sequences of individual humans. Results are open access in both project, and while HGP was explicitly designed to provide anonymity for subjects, the PGP is also a "social experiment" to see what happens when the genetic code of individuals is freely available. The community director of the latter project is Jason Bobe who also founded the most prominent network of DIY biologists[79].

Interestingly, the field explicitly associates itself with the prehistory of hackerdom. One figure who makes the connection is Tom Knight. He worked at the Artificial Intelligence lab at MIT since the 1960s, having designed and implemented such famous contraptions as the operating system and network cards for the PDP-6 and PDP-10 computers, the legendary LISP machines or Chaosnet (the first Local Area Network at MIT). He established a biological laboratory within MIT-AI and later founded a startup with his graduate students called Ginkgo Bioworks. The latter work is based on BioBricks, which standardise the procedures and formats for building synthetic organisms, enabling collaboration between citizen scientists and allowing for material and equipment to be shared cheaply. In an introductory note to the website, Jason Bobe, founder of DIYbio, recounts how the network was founded in a pub near MIT. He quotes Mac Cowell, now member of both the MIT-AI biolab and Ginkgo Bioworks, who refers to the Homebrew Computer Club explicitly as an inspiration and aspiration of the community:

> In the packed back-room of Asgard's Irish Pub in Cambridge, a diverse crowd of 25+ enthusiasts gathered to discuss the next big thing in biology: amateurs. Mackenzie (Mac) Cowell led-off the night with an overview of recent history in biological engineering, and asked the question: Can molecular biology or biotechnology be a hobby? Will advancements in synthetic biology be the tipping point that enables DIYers and garagistas to make meaningful contributions to the biological sciences, outside of traditional institutions? Can DIYbio.org be the Homebrew Computer Club of biology? (Bobe 2008)

The DIYbio network quickly penetrated the hackerspaces scene, with several hackerspaces I follow setting up various biological projects in London (Hackbase), Prague (Brmlab) and Budapest (H.A.C.K.) participating. The more hackerspace oriented Hackteria network was founded soon after, helped for instance by Marc Dusseiller touring hackerspaces and similar projects in Europe with a small biolab

[78] Specifically, the National Institutes of Health and Department of Energy.

[79] http://diybio.org/

in his car. Separate dedicated biolabs working according to the hackerspaces model sprang up soon. La Paillasse in Paris started in a squat like hacklabs do, and moved only years after, when it received official funding from the city. But the full cycle was run in Cork, Ireland, where Cathal Garvey was involved in the creation of the local hackerspace/makerspace Nexus Cork (2010), Forma Biolabs (2014), and eventually Indie Bio synthetic biology startup accelerator that shares offices between the Irish city and San Francisco (2015). (Kosner 2015)

The paradox of this latest wave of technologies absorbed in the hackerspace scene is the following. On the one hand, it already enjoys the attention of the state and capital, with the FBI officially engaging with DIY biologists since 2009 and VC funding offered to anyone with as much as an idea. On the other hand, the scene have not sufficiently transformed the technology at hand to be valorised by spectacularly successful startups, major corporations, or the military itself. For instance the involvement of biohackers brought down the price of PCRs (polymerase chain reactors) – the first machine to use for DNA testing and manipulation – dramatically from thousands of dollars to hundreds (OpenPCR)[80] or even tens (Agrawal et al. 2007), but there are no highly visible, large-scale deployments of the technology right now. Obviously, DIY biology is still in the works. Questions remain about the openness of DIYbio products too: for instance BioBricks propose an engineering solution to simplify and therefore democratise the DNA design process by using prefabricated parts without extensive knowledge of their inner workings – which sounds like the idea of the black box proposed as an innovation in open source biotechnology.

As in previous examples, it is apparent that hackers' involvement was sparkled by some very precise and highly visible advances in official science (genes), industry (printers) or warfare (drones). The net effect of hackerspaces have been to drive down prices and produce a colourful imaginary around life sciences in general and synthetic biology in particular, putting wider non-specialist audiences in contact. Given the controversial image of synthetic biology, hackerspaces could play a key role in pacifying the attitudes of wider audiences for instance towards genetic manipulation.

I tried to demonstrate through three examples of technologies incorporated into the repertoire of hackerspaces that they absolve technologies from the military-industrial-academic complex and transform them to their own image, after which innovations get absolved into the wider world ruled by the state and capital. Notably such process is similar to cultural production where subcultures pick up and find other uses for instruments, some of which then goes mainstream. The record player in hip-hop culture is a prime example of such process, but the skate board have a similar history as well.

[80] http://openpcr.org/

The moral of the story echoes the conclusion to the paper *Shared Machine Shops as Real-World Laboratories* (Dickel, Ferdinand, and Petschow 2014), where it is argued that contrary to established discourse on "the coming revolution" (Anderson 2014) or "the second industrial revolution" (Gershenfeld 2005), hackerspaces are merely small niches in the innovation ecosystem. Hackerspaces as self-organised and self-managed quasi-institutions are somewhat sheltered from economic pressures, institutional agendas, and the harassment of authorities. These conditions are ideal for experimentation with technologies that doesn't require high initial investment, including building cheap and simple versions of existing technologies, while spinning social imaginaries around them which reframe their use cases.

A tentative hypothesis is that hackerspaces actually transform selected technologies from the Fordist to Post-Fordist paradigms. They take them out of the factory, army or academic contexts which are highly institutionalised, hierarchical and capital intensive. They organise them based on self-mobilisation and enrolment, the project order, make general workshops turning out customised products, with mixed criteria for the valuation of results. Knowledge, know-how and open collaboration often works to replace fixed capital, large organisations and work discipline. In fact the targets mentioned above have been – in some specific ways at least – the most unaffected by the post-Fordist restructuration of economic life which took place since the 1970s: aircraft development for the army, industrial manufacturing for capital, and pharmaceutical science for academia. Arguably, hackerspaces took on the *hard cases* of post-Fordism. Ironically, our preliminary analysis above shows that instead of ushering in a new era organised according to the principles of peer production, they seem merely to supply elements to update classic Fordist production.

7.2 Hackerspaces as the missing infrastructure of hardware hacking or open hardware

FLOSS production famously requires minimal initial investment, since software, documentation and support are all available online. The hardware is usually owned by the workers, so that the capitalist don't even have to invest in it: the diggers bring their own shovels to the job. Finally, whole ecosystems of free (as in beer) services offer auxiliary services like email, project management and backups, etc.

These advantages are also true for individuals: given some cultural capital, a laptop, and connectivity, it is not that hard to learn enough to make simple contributions to the field. Making hardware is an order of magnitude more difficult, which is still not prohibitively hard. A physical laboratory and more haptic knowledge is required. The latter is about physical manipulations which is hard to communicate and even represent symbolically (for instance over the Internet).

The FLOSS movement – including small companies and big corporations – has worked for decades to build this infrastructure and make it available to anybody who wants to appropriate it. Much of it is useful, and indeed necessary, for building hardware too. However, hardware requires the above subjective and objective components to build. Hackerspaces function as a missing piece of the puzzle. One can enter even without paying a membership fee, find basic components and necessary tools, as well as help with the implementation. It seems that (except perhaps in Southern European countries) such an infrastructure is already in place in most European capitals, and larger cities in the US too. Hackerspaces complement online documentation, downloadable designs and forums with their physical counterparts, which enables open hardware production for anybody sophisticated enough to seek out help.

Of course the support for building open hardware has its limitations too. It is a common misunderstanding – online in FLOSS and offline with open hardware – that the people you find in a hackerspace will build for you whatever you want for free. This is not the case: people who want to build something have to demonstrate initiative and the ability to learn. If the project is inspiring for others, collaborators are found and it can even became a flagship project of the hackerspace which involves most participants. On the other hand visitors who want to exploit the free labour of hackers are often disappointed. Some hackerspace homepages list things like "I think my fiance is cheating on me again. Can you help me hack into her email account?" in their Frequently Asked Questions (FAQ). (HackerspaceSG 2014) I've heard in hackerspaces in Amsterdam and Budapest the saying popularised by the Anonymous hacker group "We are not your personal army." in response to such requests.

There is a wide-spread consensus in the scholarly literature that shared machine workshops – including hackerspaces – answer to the particular infrastructural needs of OSHW hackers and further up the line, biohackers. This underlines the importance of hackerspaces as part of a movement for "infrastructuring peer production" (Kohtala and Bosqué 2014). Troxler (2010) writes that labs are "primarily offering infrastructures", serving as a "practical infrastructure and means of sharing projects". Kostakis, Niaros, and Giotitsas (2014) argues that the peer production of tangible artefacts requires a tangible infrastructure, noting that the "hackerspaces studied provide different degrees of access to infrastructure". In line with their argument, Siefkes (2011) observes that "hackerspaces and Fab Labs are the first forerunners of a commons-based production infrastructure".

In her analysis of global collaborative innovation regimes in synthetic biology and nanotechnology in the hackerspaces, Kera (2014) extends the understanding of infrastructures to knowledges, writing that the hackerspaces are "making available less expensive laboratory protocols and infrastructure" for participants. Further expanding the understanding of infrastructures, Seravalli (2012) adds that infrastructures should be understood in a relational way including the social relations that they enable and have built up over time. In a similar vein, Moilanen (2013) observes that the "infrastructure of cooperation" in shared machine shops includes the social network of members who have a working relationship with industry actors such as manufacturers (for instance through a day job or consultancy). Silvia Lindtner and her co-authors continue to document how participants in the Shenzen innovation networks have been initially confronted with a lack of such infrastructures and how they have established them to enable collaborative production practices (LindtnerLi2012a; 2014; Lindtner, Hertz, and Dourish 2014).

In conclusion, hackerspaces can be seen as the logical continuation of FLOSS infrastructures which necessarily extended to physical locations and embodied communities.

7.3 The modern institutional grid

In previous sections I have shown processes of institutionalisation ontogenetically in the hacker scene as a whole (in the 1990s) and philogenetically in shared machine workshops (from hacklabs to hackerspaces after the dot-dom boom). Here I would like to point out the opposite: how hackerspaces elude the division of labour between the elements of the modern institutional grid. The modern institutional grid enforces a division of labour between categories of institutions. The most relevant ones here are education in the school system and the lower tiers of the academia; research in specific research centres and the higher tier of academia; production in the private sector. The role of civil society is rather vague in this regard, mostly relegated to the provision of services which neither the state (the public sector) nor capital (the private sector) can provide, as a sort of countermeasure to the other two. Arguably, this is the very reason why hackerspaces choose associations and foundations for the legal implementation of their organisational models.

It would be a mistake to simply analyse hackerspaces as civil society organisations (CSOs) though. Firstly, most available tools are geared towards Non-Governmental Organisations (NGOs), which are a particular kind of CSOs. Secondly, I argue that the social dynamics of hackerspaces draw less from the ethos of civil society then the semi-autonomous culture of hackerdom, therefore the explanations which CSO- or NGO-tied methodological frameworks can offer are of limited value. Unlike NGOs, hackerspaces are almost self-sufficient financially, and their agendas are not really shaped by available grants and master narratives set by powerful donors. Neither do they conceptualise themselves as counterweights for the inefficiency of state and capital or even liberal democratic capitalism. On the whole, the primary mission of hackerspaces is not to advocate this or that cause, nor to influence public debate or the views of the general population.

Not that hackerspaces are not performing any of the functions of CSOs or NGOs: indeed, as we will see in more detail later on, in many aspects they are doing so much more efficiently than those entitled organisations. It is only that they operate in a different framework. This point is hard to make clearly, since it rests on subtle qualitative differences in organisational dynamics, culture and habitus. Let it suffice to say here that they cultivate a different relationship to their institutional environment then most CSOs.

In fact, my claim is that institutionalisation, rationalisation and commercialisation as shown in the preceding sections have been counterbalanced with the development of a semi-autonomous engineering culture in the hacker scene at large.

While formally institutionalisation supported the development of hacker culture, in many respects it went against it, answering to pressures by more powerful institutions. The latter pressure ranged from downright prosecution of hackers (see Operation Sundevil versus the EFF) to more subtle environmental factors that any organisation working with the private sector within the confines of the open market have to deal with (like the necessity to have a legal entity for signing the rent contract for a hackerspace). As argued above, the parallel processes of institutionalisation of the the scene and the development of a semi-autonomous culture resulted in hybrid institutions which could negotiate and if necessary, translate between subcultural attitudes and mainstream normativity.

Therefore, in the particular case of the hackerspaces, the foundation or the association is merely the institutional form to which the social content of hackerspaces could be translated. In other words, the official organisational titles do a good job to describe the dealings of the hackerspace with state regulations or market operations, but fall short of capturing the social relations inside the hackerspace itself. Admittedly, such tensions between form and content are relatively common in the world of civil society – an overloaded term that has to account for the greatest variety of phenomena.

The relevant aspects of the modern division of labour are the following. Institutions are compartmentalised according to three different functions: education, research and development, as well as production. Education is undertaken in school and graduate courses. Graduation marks the point where the student proves that she has acquired expertise in a field. Undergraduates are not expected to do and often actively discouraged from doing original research. Doctorate marks the point where the candidate is able to contribute to that field. After the post-doctoral phase researchers can actually undertake more ambitious projects which they were not allowed to do earlier. In order to go beyond prototyping and design something actually useful, they most often have to move into the private sector, however. An academic institution can sell educational packages like courses and patents, but not microwaves or lasers, for instance.

For the same reason, some especially successful individuals could be in a graduate programme to study, in a research centre to contribute, and in a company to produce actually useful results. While the clear separation of concerns and the sharp focus on core missions ease the management of these institutions, the underlings are alienated from their work, and consequently from the institutions themselves, because they cannot follow the leads offered by the subject matter at hand. Dabbling into any field on any level of expertise can lead to various things all at once. Once it requires learning, another time developing on the state of the art, and then sometimes it leads to the possibility of producing something directly useful. There is no single structure which can facilitate even these three different things that can come out from a curious mind looking into an emerging field.

Similarly, leading organisations seek to bring down the barriers between education, research and production. Students are engaged in project-based learning. Institutes allow sabbaticals to their researchers. Corporations pay for further training of their employees. However, as long as these activities remain exceptions to the rule, they have to be themselves compartmentalised *within* the organisation, and justified by its proper goals. Student are allowed to do projects *because* they are supposed to learn from it, not for their use value. Sabbaticals are allowed for learning new things *because* it makes them a better researcher, not simply more educated. Corporations argue for training employees *because* it would increase profit, not because it is a meaningful human activity, for instance.

Division of labour through strict compartmentalisation is actually a problem for contemporary regimes of capital accumulation: reform proposals like "life long learning", "interdisciplinary research", and "social enterprises" are only some of the concepts that seek to address them. It should be observed that all three concepts seek to take a single compartment (education, research and production) and stretch it as far as possible, rather than escape the modern matrix in which these activities are separated. As long as explicit legislation holds the bar between education and research, or research and production, it is hard even for progressive capitalists to permeate them.

None of these prospective solutions confront issues with such gestures of radical refusal as the hackerspaces. Members often come to hackerspaces motivated by a deep disaffection by one or more modern institution: university students feel that real knowledge can be gained through hands-on experimentation; researchers that specialisation restricts creativity; engineers that the proper implementation does not contribute to the bottom line of their companies. In short, students want to be researchers, researchers want to be students, and corporate workers want to be craftsmen. Hackerspace participants are by definition learners, researchers and producers, so the tensions of compartmentalisation are overcome. Therefore these boundaries are not even articulated properly in the guise of the institution or on the level of everyday practices. In fact, the reinforcement of compartmentalisation depends on hierarchical dualities such as student/teacher, candidate/supervisor, engineer/manager — all of which are incompatible with peer production practices and therefore exempt from hackerspaces. As explained in the theoretical framework, peer production as an organisation of labour does not mean a lack of coordination or a lack of authority, but tasks and resources are not distributed through chains of command like in modern institutions. As we shall see in the case studies, it is usually impossible to determine if a nascent project is mainly an educational, research or production oriented. Indeed, the average hackerspace participant may very well be indifferent as to how it plays out.

Once again, hackerspaces are not really "new" or "emergent" formations in the sense of the untimely: they are squarely situated in social history and their subversive edge is the result of their spearheading of the latest accumulative regimes. Conflicts between the old and new regimes, and often between the actual capitalist fractions, their associated elites and other social groups, can be easily misread as revolutionary fervour. In the most ironic cases starry eyed activists fight against what they perceive as global capitalism while in practice they are merely ushering in its latest update. Having said that, such transformations by definition hold a genuinely subversive potential, which is sometimes actually exploited in hackerspaces. Hackerspaces don't enforce a division of labour greater than what is implied in the hacker moniker, thus make it possible for their members "to do one thing today and another tomorrow, to hunt in the morning, fish in the afternoon, rear cattle in the evening, criticise after dinner, just as I have a mind, without ever becoming hunter, fisherman, herdsman or critic." (Marx 1845 Part I: Feuerbach.)

The lack of an explicit division of labour in the hackerspace undoubtedly results in less alienation. Interestingly, this point is more appreciated in some hackerspaces than in others. All except one Dutch hackerspace have boards with members who enjoy extra privileges and take on more responsibilities, while in the American reception of the hackerspaces idea, "community managers" can get a salary for ushering members around. Both practices are justified by hackers as pragmatic solutions to avoid "drama" and enable them to concentrate on technical work. It is useful to contrast such an outlook to the anarchist or autonomist political culture of hacklabs, where any divergence from anti-authoritarian assembliarism is clearly marked as reactionary: the hacklab is held up as a proof that people can govern themselves without institutions. But alas, a hackerspaces is more often than not a registered institution, and its inspiration is as revolutionary as any gentlemen's club's.

Hackers overcome compartmentalisation through a profound, almost mercurial lack of interest for the imaginary ontological barriers separating these functions, following their curiosity wherever it takes them. Söderberg calls it *play struggle*, combining resistance to social pressure with the joy of discovery (Söderberg 2008). Coleman calls it the *trickster*, breaking social boundaries since times immaterial (Coleman 2012; Coleman 2014). It could be all that, but I argue that in the final analysis it is the recognition of life, and in particular the dialectical subject/object interaction, which in all its complexity comprises a single phenomenological unit. That is, it is possible to break down interactions between living and non-living matter into different modes and relegate them to separate institutions, and instil a habitus that is not sensitive to the multiplicity of interactions, but these attempts at social engineering deny the structure of the actual phenomena they are dealing with. Therefore, complication, contradictions and frustrations ensue.

In a hackerspace the only required reason for implementing a technical idea is simply that somebody is willing to work on the project. In the same way that anybody with the tools and the knowledge are allowed to write (FLOSS) programs, people can build contraptions in hardware at the hackerspaces. It is often unclear if the project will turn out anything significant at all – perhaps it is merely a futile stab at an obscure problem. Similarly, it is often undefined whether the actual project falls into the domain of education, research and development, or production. Once definitions are established, they can also completely change through the life time of the project. At best, what started as an attempt to learn electronics can lead to an advancement in engineering which could be turned in to a mass manufactured product (see the first case study in the next chapter). At worst, the development of a product could turn out to be inviable but perhaps at least a contribution to the field, and end up as a frustrated attempt at learning something new.

Between these two, a "dual-use" technology class often found in hackerspaces is the *kit*. A kit is a handful of components coupled with instructions on how to put it together. They are typically simple and cheap, as well as easy to assemble, yet they demonstrate an engineering trick (a hack) which makes them nifty. Many hackerspaces produce kits because they perform three functions at once. First, making and selling kits can provide a source of complementary revenue for the hackerspace. Kits are produced by the dozen whenever they run out and taken to events, fairs, and offered to visitors on site. Second, kits are good for teaching novices and therefore improving the skill level of participants and visitors. Putting together a kit means learning a basic skill and realising a counter-intuitive truth, all while producing a tangible result which can be shown to others as a badge of proof. Third, kits advertise the hackerspace and showcase the ingenuity of its members. Kits can get far from the hackerspace and carry its story with them. Evidently, they seamlessly combine at least education and production (and in this case, marketing). In comparison, in case a secondary school wanted to do kits that students can use for project base learning in physics classes, it could not. There would have to be a supplier company which does the manufacturing, and the school would have to acquire the product to use it for education. Needless to say, the kit would be harder to adopt to local needs, resources and culture in such a complex institutional environment.

Two examples are worth mentioning here. One is the Joule Thief available at the Hungarian Autonomous Center for Knowledge (H.A.C.K.) in Budapest. (H.A.C.K. contributors. 2011) It is comprised of half a dozen components which can be combined to suck energy from a standard AA battery which has already lost its capacity. The contraption uses the last fractions of discharge to light up an LED for as long as a month. Anybody wanting to learn how to solder enjoys the exercise, but it is especially popular with children. They wonder how is it possible to use a battery which is already useless in normal devices - the answer is of course in the way it is wired up to the LED. The printed instructions and the components are given away in a small plastic case, for instance to low skill visitors who come to the hackerspaces without a specific objective.

Another example is the TechInc logo badge sold in Technologia Incognita, the hackerspace in Amsterdam. It is a printed circuit board with the logo of the organisation on it. The logo of the organisation looks like a classic steering wheel for a ship, but it is actually a working circuit design for lighting up multiple LEDs using a single power source. The design itself demonstrates a little known engineering trick for distributing power to multiple sources. Once the lights are in place the final step is to install a battery on the backside of the design which lights them up. The kit teaches soldering to novices, but also a trick of the trade to more experienced hardware hackers. As a sort of merchandise, it provides revenue for the space while reminding visitors who take it away to the good times they have spent in the hackerspace.

Andrew Pickering asks “Where might an alternative to modernity flourish?”. (2010, 400) In response surveys an array of historical organisations from the 1960s and 1970s – some realised, some planned – which elude the matrix of the modern institutional grid. The Kingsley Hall and Archway anti-therapy communities of R.D. Laing (186-197) are good examples: the dichotomy between patients and doctors, normality and pathology would be confused, to create “a place of reciprocal transformation for the mad and sane alike: 'This would appear as ex-patients helping future patients to go mad.”' (199) Psychiatrists lived together with their patients and volunteers of various ranks and kinds, going through everything that a psychotic episode could mean in the context of a community. Learning about madness, developing one and healing have been only three functions of such a community space. In Kingsley Hall anybody concerned with altered states of consciousness – or *strange performances*, as Pickering puts it – were welcome to develop their interests. It is easy to see how the Kingsley Hall and Archway communities overstepped the institutional limitations of contemporary psychiatric wards through following the various impulses which could make madness a productive force of positive experiences.

Another experimental institution was the Fun Palace planned by the neo-futurist architect Cedric Price, conceived and commissioned by theatre director Joan Littlewood with help from cybernetician Gordon Pask. The idea was to create a sizeable building for mixed use by the public, including relaxation, ateliers and performances. The undefined use of the building would be catalysed by undefined space: movable walls, floors and roofs that encourage experimentation, adopt to unknown future requirements, and lend themselves to temporary schemes. Needless to say, vigorously opposed by the local government, churches

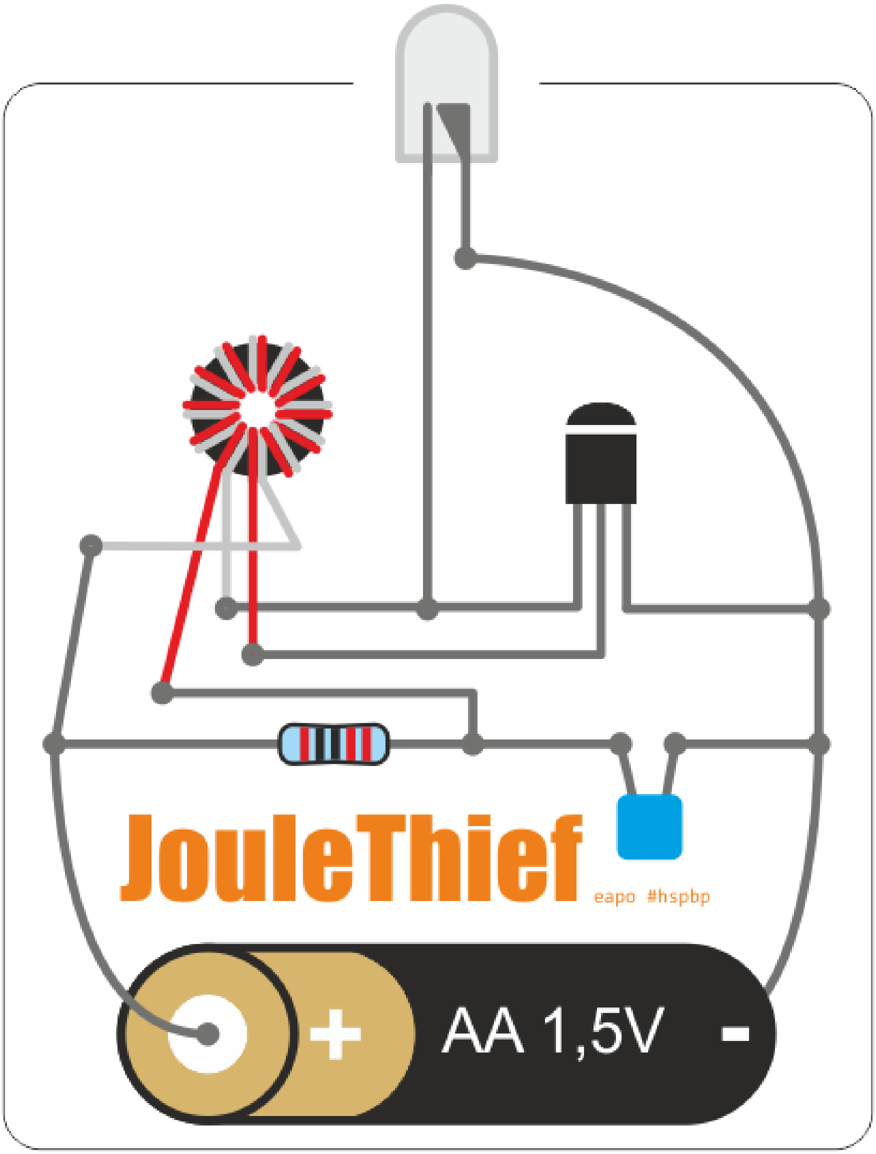


Figure 12: Joule Thief kit circuit design from H.A.C.K. (Budapest). (H.A.C.K. contributors. 2011)

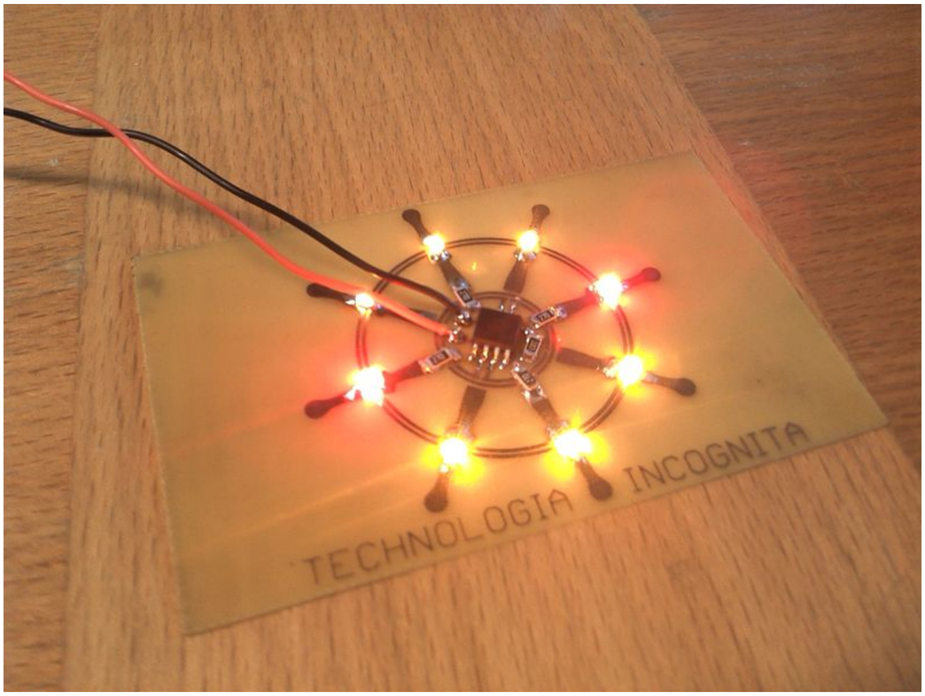


Figure 13: TechInc logo PCB kit from Technologia Incognita (Amsterdam). (Brainsmoke 2014)

and conservative architects, the project never got a planning permission and eventually faultered. The very indifference showed by the advocates of the project for situating their venture in the modern institutional grid doubtless contributed to its demise.

I suggest that hackerspaces fall into the same tendency, going beyond – or rather between – the school, the research centre and the factory in their blatant disregard for the institutional limitations imposed by the compartmentalisation that ensures the division of labour in modern society. As spaces of freewheeling, undirected technological creativity leads to a difficulty similar to that experiences by Pickering's anti-institutions based on the amodern cybernetic ontology. It is hard to conceptualise, conceive and sustain them in an institutional environment which favours clear-cut functions and determined outcomes. Pickering makes the point that one reason for the strand of amodern cybernetics he explores to have effectively died out was that it was unable to "reproduce its social basis", meaning the institutions which could support and the social groups which could practice these activities. His conclusion is that if the implementation of an amodern engagement with matter have been generally thwarted by the official institutions, a parallel "underground" world could serve as a home for them.

Hackerspaces fulfil such a function of hosting and cultivating forms of life founded on a practical engagement with technology. Significantly, both hackerspaces and Pickering's experimental institutions found the most support in the sphere of art and education. Art and education are notoriously process-based areas where undirected creativity can find its outlets. Support is often available for ambiguous projects due to the lack of a stable definition which would safely circumscribe these spheres. More than the practitioners of most other occupations, teachers and artists are taught to work as with their whole personality as an integrated human being, and work on others as single human beings too. Therefore these spheres are more likely to be able to incorporate amodern tendencies which address forms of life as a whole. Luckily, while many artists and educators are enthusiastic about hackerspaces, they are not sustainable because of arts or educational funding. As we shall see later on, the key to their sustainability is a core of highly privileged, ideologically overpaid workers with available leisure time. Hackerdom as the "parallel world" of playful tricksters is perhaps one of the few milieus where the amodern tendencies identified by Pickering can reproduce and expand their social basis.

It is interesting to note that despite their disparate inspirations, all these anti-institutions have a tendency to become "community centres" of sorts.[81] A common drive behind anti-institutional efforts is to restore the fullness of life without alienating functional specifications which delineate the proper use of spaces, thus dividing the life world into spatially distinct arenas all with their specific socialities. Members do come to hackerspaces to tinker on projects, but sometimes they just come to have a chat, eat a pizza or have a shower – or even

[81] Kohtala and Bosqué (2014) show how one of the first Fab Labs established in Europe became much more of a community centre than a technologically active shared machine shop.

to escape their work- and family lives. The following section examines the last imperative.

7.4 Third spaces

Schrock (2011) rightly observes that in the context of contemporary urban life, hackerspaces function as *third places* (Oldenburg 1989): participants use them to socialise away from both their work and domestic environments. Putman (2000) argues that third places are essential for a healthy society where massification and commodification results in rampart alienation, questioning the basic solidarity between people sharing the same territory and tearing the social fabric of society. His critique is based on a historical survey of clubs, lodges, building societies and similar associations where citizens cling together according to loose principles that can cut across professional, class and gender lines. Putman has a particular concept of *social capital.* He mainly uses the term to refer to connections between people who share neither familial nor labour ties, connections between people from different classes or neighbourhoods. The role of social capital is particularly important in cities which are by definition full of strangers. In order to feel at home in the city, one has to maintain contact with people who have different lives and perspectives from oneself. These connections are essential when it comes to "organisation" in any sense of the world, but particularly organisation around social problems. Indeed, Putman identifies a whole range of social problems which get worse if societies loose social capital, as did the US in his argument:

> Over the last three decades a variety of social, economic, and technological changes have rendered obsolete a significant stock of America's social capital. … Our growing social-capital deficit threatens educational performance, safe neighbourhoods, equitable tax collection, democratic responsiveness, everyday honesty, and even our health and happiness (367).

It is interesting to see how some of the same social, economic and technological changes which have rendered social capital obsolete also lead to the rise of the hackerspaces which arguably address these problems from an up-to-date vantage point. Of course this is not to present hackerspaces as the ultimate solution to the problem of alienation in capitalism and the massification of urban life, since without doubt they gather a very particular social group whose members enjoy various privileges (as detailed in the next section). However, the story about the organisation of the hacker scene and the establishment of the hackerspaces echoes the ameliorative recommendations issued by Putman in the last two chapters of his book (287-444). Similarly to arguments about virtual communities and social networks, hackerspaces can be seen as attempts to rewire the loosened social fabric using ICTs. As I try to demonstrate in the following paragraphs, the embodied community life in hackerspaces more closely resembles the old models then its exclusively technologically mediated counterparts.

The quasi-institutions described as third places by Oldenburg (1989) are not exactly the associations that Putman is focusing on, but perform a similar function, and answer to the same problem. Oldenburg is writing about the coffee house, the barber shop, the community centre and similar places at the border of private and public spaces. These address the problem identified by Putman, e.g. the lack of cross-sectional connections between mostly urban populations.

The characteristics of the "great good places" that Oldenburg identifies fit neatly on hackerspaces. In the following paragraphs I recap his analysis applying it to hackerspaces. At the same time, I show how hackerspaces also diverge from that pattern, since rather than harking back to the good old times they are addressing contradictions at the cutting edge of capitalism. Finally, I point out how these same characteristics differentiate hackerspaces from the next waves of shared machine shops, which is particularly significant since otherwise hackerspaces and the later genres are often put in the same theoretical bag by analysts.

Third places function as a *neutral ground* for social interactions where differences between actors are not marked out so clearly as in the world of work or family life. "Going to drink with my supervisor" is an awkward situation exactly because social interactions between clients in a pub happen in a lateral way that subverts the clear hierarchy that the word "supervisor" marks. "Smoking a joint with my mother" presents the same pattern. Hackerspaces provide such a neutral ground. Anarchist activists, happy or disgruntled corporate employees, and military contractors can discuss matters of technology as much as politics, as it happened in the Budapest hackerspace during a lockpicking workshop. Another such instance was during a visit to the London Hackspace where the three rooms hosted very different conversations. One was an operative meeting to manage the technical infrastructure of the local Occupy movement, the other an experiment with hypnosis and the third a conversation about the new CB radios that somebody just installed for the London police. These places therefore function as a *leveller*, connecting people from different walks of life who would normally not meet, or meet in a situation which is much more symbolically overcoded or choreographed.

For Oldenburg, a necessary but not sufficient condition for such social interactions is that *conversation should be legitimate and central part of the activity* in third places. Results of quantitative research like Moilanen (2010) support the hypothesis that hackerspaces are primary used for socialisation, and while hacking, or the research and development of technologies is central to hackerspace life, it is sometimes not more than a social lubricant. Indeed, next to pure conversation, the very practice of collaborative technology development provides ample opportunity for mutual engagement between parties who would otherwise not meet. Another crucial aspect outlined by Oldenburg is *accessibility and accommodation*, which have already been discussed: the open door policy of hackerspaces make them rather accessible to the general public and their ubiquitous sofas accommodate conversationalists and technologists alike. Furthermore, the author identifies the presence of *regulars* as yet another condition, which is neatly provided by the membership model of hackerspaces. At this point the main thrust of Putman's and Oldenburg's arguments intersect: hackerspaces are both civic associations organised around an ethos wielded by a particular social group like in many of Putman's examples, and socially under-determined physical places away from work and home as in Oldenburg's ideals.

Perhaps a digression would serve well to further illuminate the points above. One of the most fascinating realisations of my field surveys in hackerspaces was the sheer amount of time members were willing to spend explaining their lives and contraptions to me, a random anthropologist who stumbled into their club. It was a common experience that upon a few hours' notice, one or two members would be happy to spend the better part of the evening with me, putting aside any other responsibilities they had or projects that they were working on. The practice fitted into the regime of leisure reminiscent of a coffee house more closely than the machine shop suggestive of disciplined work. It supported the hypothesis that the hackerspace is mainly a site of socialisation, and that regulars are open to random encounters with one-time visitors.

The last set of criteria Oldenburg sets out are more subtle, having to do with the atmosphere of third spaces. A *low profile* contributes to accessibility and a neighbourhood clientele. Cool cafés with their trendy decors attract tourists but the inglorious corner bar is where the locals hang out. Similarly, most hackerspaces are virtually unrecognisable from the outside. They do present a fascinating image inside, but that sort of aesthetics have little to do with any conceivable conception of interior design, and much to do the sort of affectionate clutter that one finds peering inside the cubicles of office workers. Posters, notes and of course gadgets have more significance to insiders than outsiders, and generally pile up rather than arranged. Here however it is worth to take note of a significant difference between olderburgian places and the hackerspace model. The attraction of the former is primarily territorial while the attraction of the latter is primarily thematic. Of course the users of any hackerspace mainly come from the vicinity: the city circumscribes its membership. However, the first principle of composition is technology enthusiasts rather than locals from the neighbourhood. The much rehearsed argument that the Internet allows people with common interests to come together and special interest groups to organise themselves more effectively – all in the face of heightening alienation and increasing social isolation — can be easily applied here.

A *playful mood* that Oldenburg ties to third places is another characteristic of hackerspaces which is glaringly obvious in the first impressions of visitors, in the self-presentation of members and in the scholarly literature of the scene too. In the spirit of the hackerspace slogan "Be excellent to each other"[82], visitors are greeted by a jolly pack of hackers and an informal atmosphere. It can be no doubt intimidating to many, for it is not apparent how to join the conversation in case one is not well versed in the sometimes highly technical topics under discussion, or if members are engaged in deep concentration on

[82] From the influential Noisebridge hackerspace.

their projects and only exchange fleeting remarks. However, once hackers start to show off their creations and the quirky logic of their machines in a mix of bragging and self-depreciating tone, it quickly becomes a sarcastic commentary on hegemonic attitudes to technology. Secondly, checking the mission statements and definitions on hackerspace websites already makes it clear that they are places for unalienated labour and undirected creativity, because the emphasis is on the availability and accessibility of tools to "make almost anything". In this regard technological tools are fashioned as toys to play at your pleasure, to realise your dreams, and not precisely as part of the puzzle in the construction of useful engineering solutions. While in the previous sections I argued that hackerspaces mix education, research and development as well as production, their self-presentation hints more at play than anything else, which is probably true of the whole DIY ethos.

Finally, the literature on hackers speaks volumes on their playful mood. I have repeatedly referred to Söderberg's concept of *play struggle* and Coleman's concept of the *trickster*, both of which seek to capture the jocular essence of hackerdom. Coleman (2012) in particular focuses on the idea that the structure of the joke captures the hackers' relationship to technology. Hacker culture is arguably organised around and transmitted through in-jokes, which calls attention to particular examples of the hack, selects what is important and what is not, and provides a practical way of transmitting engineering attitudes in an informal manner to new participants. Three good compendiums of hacker humour are the Jargon File (Raymond 1992) that collects language from the MIT/AI era, the series of joke RFCs (Request for Comments) which normally define Internet standards[83] and the XKCD web comic that can launch whole subcultures or technology best practices with a single drawing of stick figures (Munroe 2005). Finally, searching for the sources of pleasure in engineering work – which commentators agree are deep and diverse –, Kleif and Faulkner (2003) concludes that the most fundamental engineering pleasures experiences by software developers and robots builders alike[84] is the promise of engineering to overcome ubiquitous uncertainly and lend a sense of control to the makers of technology. Under these circumstances, they argue that both professional and hobby engineers use play as an attitude and a tactic to defend them during the build process from the ontological uncertainty inherent in the construction of stable technological artefacts. Of course hackerspaces explore such aspects from a freewheeling perspective since they are there for technology enthusiasts to band together and celebrate their culture.

"Home away from home" is the last theme explored by Oldenburg, and already discussed apropos the HackBase "sleeping in the hackerspace" tendency where

[83] "Although a few RFCs contain humorous portions only, the vast majority, most of which are coincidentally dated on the 1st day of the month of April, are funny in their entireties. To us, one of the most notable is RFC 2324, the Hyper Text Coffee Pot Control Protocol (HTCPCP/1.0), which pokes fun at HTTP (a real protocol that has become hugely popular)." (Inter-Corporate Computer & Network Services, Inc. 2012)

[84] A fairly good approximation of hackerspace members.

the hackerspace feels so cosy that there is an urge to transform it into your primary home. Here it is worth to refer to the idea of the hacker as the ideal post-Fordist worker: a form of life featuring total productivity. A rather caricaturistic characterisation of the hacker figure quickly reveals the hacker as a victim of the Protestant ethics on steroids. Hackers tend to work for free, around the clock, exchanging leisure time and recreation for more and more work, which they perceive as their principle source of pleasure, e.g. play. Kleiner (2010b; 2010a) points out precisely that FLOSS is primarily good for capital simply because it can be freely appropriated: a commons prepared especially for primitive accumulation. Prototypical hackers tend to neglect even the meals, showers, cleaning and family life in addition to sleeping as they became the kamikaze engineering — a worker who does not even reproduce itself, but finds everything needed in the world of work itself. Of course this is in stark contrast to Himanen (2001) who argues that the hacker ethics is the negative blueprint of the Protestant ethics defined by Weber (1958) as the ideological basis of capitalist workers' discipline. Of course such a totalitarian and fundamentalist look at hackers does not do justice to the complexities of hackerdom or the nuances of hacker practices. For one it should be mentioned that the tendency above means that hacker culture incorporates a rich set of carefully tailored practices of cooking, cleaning, caretaking and managing personal relationships that bring many aspects detailed above into the sphere of work. In the final analysis however, we can pronounce that for better or worse, hackers have a tendency to collapse work and other aspects of life in a process where whether work or life comes out on the top is incidental.

Indeed, my own observation in London (2011), Hamburg (2013) and the Netherlands (2013) during Christmas eve show that members who cannot or choose not to be with their families during this time meet in the hackerspaces to enjoy each others' companies, and occupy themselves with tinkering. I guess that a place where one can spend Christmas even in peace and harmony qualifies by definition as a *home away from home*. Yet this observation also calls challenges the tripartite division of life that Oldenburg and Putman build their analysis on. As the Christmas eve example shows succinctly, sometimes third places are not simply a complement for family and work life, but even a substitute for it. People without close family ties, enstranged from their "loved ones", and perhaps without work or at least a workplace, can still find a meaningful social context in hackerspaces. Such patterns are of course more often observed in Eastern Europe where precarity in particular and economic problems in general cut deeper.

As I hinted above, hackerspaces can be cut off using these criteria from other genres of shared machine shops. This obviously contributes to a better understanding of hackerspaces and their particularities in the DIY field or the ecosystem of peer production phenomena. Interestingly, in the above respects hackerspaces arguably resemble the neighbourhood barber shop more than the Fab Lab at the local university campus, even if most scholarly works would treat them as the same thing. Shared machine shops tied to institutions typically have a more stratified membership model than hackerspaces. In the latter case, a typical European hackerspace have nominal roles for a board of directors, members and visitors. As stressed above, these nominal roles are often implementation details of the legal form whose social content does not necessarily correspond to them. For instance in RevSpace (Den Haag, Netherlands) a board member explained that they agreed on the strategy of actively trying to minimise the differences in what a member and a visitor can do, in order to make access to the hackerspace more open and democratic. Generally the most significant difference is that visitors have no key to the space so that in practice they can only be there under the supervision of members. In the hackerspace in Budapest, Hungary (H.A.C.K.) the privilege of members in addition to holding a key is to be able to get Club Mate 17% cheaper,[85] and to be able to rent a box for storing projects.

In contrast, the shared machine shops which are tied to institutions typically have a single acting director, several actual managers – including a community manager –, members and signed up volunteers. They often restrict access to the space for non-members during most of the official opening time. The illusion of continuity between genres of shared machine shops is best illustrated by the Open Thursday tradition, which is the day dedicated to welcoming visitors, potential new members, friends and family. In most European hackerspaces OT means that members are prepared to deal with interruption and to answer stupid questions, and generally make an effort to be sociable and interesting – for instance by staging small presentations, cooking food, or simply gathering in larger numbers than normal. This happens more or less the same way in MIT affiliated Fab Labs as well, with the significant difference that on other days of the week access is restricted to members. As already explained, hackerspaces are open to visitors any time of the day as soon as a key holding member opens the door. Under these circumstances it is hard to describe them as a *neutral ground* or a *social leveller* since the roles and responsibilities of attendants is clearly marked out. Commercial shared machine shops like some Fab Labs and all the US Tech Shops[86] are companies that rent time on a hackerspace-like repertoire of tools on a per machine, per hour basis. Therefore the community life that they are proud to foster in their public relations material are constrained by the limits imposed by the social dynamics of a photocopy shop.

As it starts to be clear from these comparisons, most next generation shared machine shops are not hangouts where *conversation is a central part of the activity*, but focus on the production of results. The quote from the mouth of a maker I heard from Debora Lanzeni that I will continue to use repeatedly in conjunction with the concept of *unfinished artefacts* is preside here: “hackers never finish anything” – so they have time for whatever they happen to be enthusiastic about. Another diagnostic difference that pertains to the *accessibility and accommodation* that hackerspaces provide vis-a-vis other shared machine shops is how training happen. In a hackerspace it is customary that people arrive much

[85] Club Mate is the default hacker drink as chronicled by Thomas (2014).

[86] Operated in a franchise model by a chain store of the same name.

earlier than the starting time, or even much later, and stay around hacking after the training is over. The more institutional next generation spaces may even close up after the lessons are over and participants would be busy rushing home anyway. Therefore they could barely be called a *home away from home.* As far the *low profile*, I will show later on in more detail that hackerspaces are generally more eclectic in their interior design, less picky in their cleaning standards, and generally worse lit.

The moral of such an overview is that while in comparison to hacklabs, hackerspaces look like the recuperated version of the original shared machine shops, yet set against the standards of new generation SMSs they are veritable community centres.

While that may sound overly modest, it is worth to remember Bifo's testimony (Franco Berardi a.k.a. Bifo, Jacquemet, and Vitali 2009). Comparing his experience in organising the industrial working class in the 1970s and his contemporary activism organising precarious artists, the main difference he finds in practical terms is the difficulty of finding a shared time and space where and when collective experiences and subject formation can take place. Hackerspaces address both issues rather effectively, combining 24 hour access and the membership model with their own brand of social technologies for coordination. As one of the informants in Schrock (2011) says, "The primary thing you get with your membership is community."

7.5 Membership

The political economy of hackerspaces is closely tied to the political economy of their membership. Therefore I try to give an account of prototypical hackerspace participants according to their economic situation and the various ways in which they contribute to the sustainability of the hackerspace. The survey is primarily a sketch that serves to illuminate the political-economical dynamics in which the hackerspace as an institution co-created by its membership have to operate, and secondarily setting the scene of human activities in which the case studies of small-scale technological artefacts in the next chapter can unfold.

7.5.1 Third spaces for leechers

Especially in countries hit worst by the crisis, out-of-work members often put in the most hours. They are the ones who clean, build and maintain infrastructures, and work on flagship projects the most – or even just keep the hackerspace open, plain and simple. They benefit from the space in a variety of ways. For one, it provides a baseline infrastructure for their daily activities. One member in Budapest told me that he has no comfortable desk at home and that is one reason why he hangs out in the hackerspace.

Another variation on the same theme are “chess players”. A founder of the Mama in Zagreb readily recognised them based on my recall of observations about the Metalab hackerspace in Vienna. They are suspiciously homeless people who use the facilities like shelter, shower and coffee machines to complement their own, and given little interest and skills in high-tech tinkering, legitimise their presence through chess playing – a game recognised and respected by hackers for its sophistication. Similar strategies and dynamics are described on the wiki of Noisebridge, the anarchistic hackerspace in San Francisco’s Mission District, ridden with the dire of homelessness.

As already highlighted, for people who lack a first and a second place, the third place gets to be especially significant. The discussions and controversies about sleeping in the hackerspace show that for some the third place can get *too* significant. Essentially, the conflicts revolve around the question if a space can serve as a third place for its members and the general public while at the same time it is the first place (e.g. home) for some others. Interestingly, since co-working from the hackerspace (which basically means running a freelance operation or even a company) is more or less accepted practice in the hackerspace scene, there are already examples when the third place can function as a second place (e.g. work space). These dilemmas complicate the status of the hackerspace which is already a bit complicated, as a privately held space for with public access.

7.5.2 Project and networking spaces for freelancers

Beyond the basic facilities mentioned above, the “added value” hackerspaces provide is a work centred yet inspiring environment, where freelancers can stay connected and collaborating with a strong network of interesting people and gadgets.

Therefore there are members who use the hackerspace as their work office because they are freelancers. Boltanski and Chiapello (2005, 104–105) identify the project order as the basis of contemporary economic activity, which can only exist given a strong network of connections. Crucially, hackerspaces combine making connections “for fun and profit”.

The qualitative increase and quantitative centrality of precarious and flexible labour is a major structural shift in late capitalism regarding employment structures, business models and modes of capital accumulation. Since the technological sector is at the forefront of these transformation, it is reasonable to assume that such changes affect technology workers even more than others. However, while outsourcing, consultation and flexible labour networks built around ephemeral projects may perform better than other solutions of the market, they have a significant disadvantage. As Coase (1937) explained at the heyday of capitalism the whole point of the firm was to lower transaction costs. Information and know-how flows more seamlessly in-house, and collaboration is easier and cheaper if it is not necessary to negotiate a contract for each transaction.

In case flexibilisation is not based on expanding the black market, it comes with increased overhead, including more contracts and increasing transaction costs. However, for some participants hackerspaces provide similar benefits then Coase's firm. Even if workers work on different projects with different networks, they cluster around specific technologies – for instance, hackerspaces traditionally have a high concentration of Python programmers – so that they can learn from each other more effectively. Asking for help is as easy as raising your head and voicing your problem. Members have usually enough time on their hands to help each other out, also because their primary purpose of attending the space is to socialise around technology topics.

Therefore, even if it is not a conscious political strategy on the part of hackers, it can be argued that the hackerspaces are an instance of self-organisation of technology workers in face of precarisation and flexibilisation of digital labour. They seek to put into operation the advantages identified by Coase in the first part of the 20th century in an environment which changed significantly since then (described by Castells). Theorists of peer production like Clay Shirky (2008) identify similar Coasian effects in open mass collaboration efforts *online*, like in the case of Wikipedia. However, what is interesting here is that workers band together to create a *physical* space and infrastructure in order to take advantage of open collaboration, and manage to exploit it even when working on different projects.

Co-working spaces – another category of shared machine shops – are founded on a business model which offers this as a service: work alone but surrounded by interesting people! Both hackerspaces and co-working spaces have the advantage of providing flexible work environments in terms of mood and scale. Creatives don't necessarily like to spend the whole day at sitting at the same desk, but here they can spend some time at a big table with many other geeks, some other in the small "chill-out room" while smoking and thinking, and more in a cosy corner with less disturbing factors. Of course hackerspaces also provide ample opportunities to find something amusing to do in the breaks too. Regarding scale, while many freelancers work alone, sometimes they need to meet with a few people or even make a meeting with a larger team. Hackerspaces and co-working spaces can accommodate these needs too. While designer yurts and chic interiors at co-working spaces attempt to project "cool", for many clients there is nothing more compelling than the hands-on, edgy workshop air of hackerspaces which is about "getting things done" instead of worrying about the cool factor. Being called a "hacker" space can help marketing. Third places apart, if the hackerspace is the flexible home for the former group of leechers, for this group the hackerspace is the flexible office. Therefore, these members have an extra interest in keeping the space tidy and paying membership fees regularly.

7.5.3 Leisure clubs after day jobs

In the middle of the strata there are people who have more or less conventional day jobs. As with many people who have "industrially relevant" hobbies, even if members have a stable day job, they can seldom develop their knowledge as fast on the job as in their "free time" they spend in the hackerspace. Their current employer often trains them to specialise narrowly, but new employment typically requires a different specialisation, and thus the ability to learn new technologies quickly. There is no place better than a hackerspace with its constant flow of technology minded visitors, endless discussions of the cutting edge in everything, and a new obsession every week. Members with day jobs are the most popular and diverse group, roughly falling into three subcategories.

(1.) Some come to the hackerspace to do something different from their day job. A good example is the audience of TOG, the hackerspace in Dublin – a city with a famously high concentration of software developers (the headquarters of Facebook and many other online media monopolies). Many members write code during the day and arrive to the hackerspace in the evening without a computer, to hack on hardware and build physical stuff, do robotics and pick locks. The switch of the door system in TOG is the single most impressive item I ever saw during my field work in twenty-odd hackerspaces. It is a huge rusty knife switch looks as if it was taken straight out of a Frankenstein movie. Flipping it sends a clear message that software development is over, and hardware hacking can begin. Having said that, it is still a door system conforming to the uniform API, so that the signal is inscribed in an online database, syndicated on Twitter, etc.

(2.) Conversely, there are members with day jobs who come to the hackerspace to do the same as at their office desk, but in a different way. Disgruntled software engineers people hackerspaces to do proper programming, outside of the constraints imposed by incompetent project managers and profit-oriented technology decisions. Disillusioned system administrators join to maintain lean information and communication infrastructures which do not have to conform to enterprise requirements and ISO specifications. Artificial intelligence researchers and robotics specialists frequent hackerspaces to work on pet projects and find an appreciative audience.

The contribution to the hackerspace of these two groups with day jobs is of course similar. They pay membership fees and arrive after 6pm, rarely staying for all-nighters. The interesting aspect is to understand the role these typically middle class members play in the political economy and culture of hackerspaces.

Due to the centrality of information and communication technologies in the current configuration of cognitive capitalism, even during the western financial crisis where academia, real estate, and even financial services sector suffered severe setbacks, ICT firms stayed afloat. Consequently, hackerspace members with a day job are still able to pay membership fees. Thus hackerspaces are different from activist groups and NGOs: the latter rarely make enough money from membership fees to rent their office space. The core membership of hackerspaces work in a protected sector and therefore these organisations are more sustainable then similar social formations. In Eastern Europe, membership fees are enough to cover a cheap rent and buy toilet paper and similar consumables. In Western Europe, membership fee often include the accumulation of emergency reserves and covers the acquisition of equipment. At the same time many machines are donations from the extended social network of hackerspace members. Members – without regard to their socio-economic background – put a lot of work in repairing, building and recycling the equipment that forms the common resource pool of the hackerspace.

7.5.4 Work and leisure, fun and profit

As I argued above,[87] hackerspaces provide a public infrastructure for hardware hackers which complements the free source code, documentation and support forums found online. In virtually any North European capital there is at least one hackerspace which is open to the public. Interested parties can find material and human resources for realising small scale electronics projects. In this regard hackerspaces can be seen to complement and extend the online commons and its collaborative potentials to embodied communities and urban areas.

Along the same lines, there are members who come explicitly with a major project in mind – like building a 3D printer (additive manufacturing machine) or developing lock picking skills (how to open a lock without key or force) – and continue to be focused on it. Some of the results are absolved into the hackerspace infrastructure and others are taken away by those who built them, even though the knowledge and the experiences remain. The bulk of members, however, are there for the surprises and both organise and engage in a wide range of activities. This can be nailed down to two reasons. Firstly, their interest is not confined to a specific technology but technology in all its manifestations. Secondly, after a certain point technology becomes a means to an end. That end is socialisation.

One of the most striking experiences in my field work is that it often happens – as in Budapest on May 10th, 2014[88] or in Leeuwarden on December 15th, 2013[89] – that I spend more than six hours a day at a hackerspace, without seeing any actual technology development taking place. I meet many people, who come and go in hectic patterns, discuss leisurely or hurriedly, bring and take stuff, use tools and feed animals. On such days technological artefacts are nothing more than social lubricants, much like the beer in pubs. A gadget or an unfinished project is put on the table between two participants, who can then go on for dozens of minutes discussing related, and then unrelated matters. Cindy Kohtala and Camille Bosqué (2014) describes similar experiences regarding their independent

[87] In the section *Hackerspaces as the missing infrastructure of hardware hacking or open hardware.*

[88] At the Hungarian Autonomous Centre for Knowledge (H.A.C.K.), the local hackerspace.

[89] At Frack, the local hackerspace.

visits to the first Fab Lab in Norway, even though they tentatively suggest that the Fab Lab lost its connection to hacking altogether, and became a community hall for visitors and the local community. Our analysis agrees on the point that established Shared Machine Workshops (SMSs) can easily become community centres in the real sense of the world: places for people to turn to with their daily problems to help technological and social solutions, but also third places to hang out in and pop in to.

Indeed, such ethnographic evidence is complemented by quantitative data: results of the study conducted by J. aka k. Moilanen (2012b) give evidence that socialisation, not technology is the primary motivation for members to attend hackerspaces. The study contends that "[t]he social factor of peer production communities seems to be the key element. The results have been almost the same for three annual surveys." The data supports the hypothesis that hackerspaces are in some respects closer to third places than next generation shared machine shops. In some sense we can say that technological productivity is merely an interesting side effect of socialisation.

Members who develop closer ties with hackerspaces sometimes become professionally implicated. Some hackers quit their jobs or change their carriers so that they can be more involved in hackerspaces and related activist initiatives. In the United States hackerspaces have community managers and other personnel, which can sometimes become a job. Anthropologists like me may be payed to hang out in hackerspaces, as well as journalists, documentary film makers, etc. However, the most common trajectory is the *spin-off company*, which starts with a prototype developed at the hackerspace. As in indicated above in the *Hackerspaces as a black box* section, the principal example is the nascent 3D printing sector which is largely based on development efforts seeded from hackerspaces.

The high profile example is MakerBot Industries which started off by members at New York City Resistor, mentioned in the section *Hackerspaces as a black box*. Some of them later joined the RepRap Foundation, a non-profit advocate of 3D printers. They went on to found their own company which was eventually brought by Stratasys, Ltd. – an established industrial company traded in NASDAQ. However, this is only the tip of the iceberg. Local companies were started around similar ideas by members of hackerspaces in Sao Paolo, London and Vienna (just to speak of my own field work experiences). In the latter cases the employees and founders stayed affiliated members and active participants in the hackerspace and the company supports the space in various ways from in-kind donations through knowledge transfer to cash.

This way the network of companies around a hackerspace can be crucial for the long term sustainability of the establishment, even though rent is traditionally covered solely from membership fees – a measure designed to ensure the independence and autonomy of the hackerspace from both profit-oriented and non-profit actors. What I call the *relative autonomy* (explained in more detail in the *Relative autonomy* section below) of the hackerspace is partly secured by the symbiotic relationship with various sectors of the technology industry. Even though it is impossible to show in detail here, I suggest that any industrially relevant subculture have to set up and cultivate such a relationship with its "parent industries", and hackers have worked out how to do it through a long and winding constitutional history of the scene. The industrial relevance of hackers' engineering culture shows in the fact that companies also had to learn how to work with them,[90] and change their attitudes accordingly.

7.5.5 Geniuses and ghost members

A curious subcategory of freelancers and day-jobbers are geniuses. Most hackers are not geniuses but hackerspaces sometimes have a few hacker geniuses lying around. They are people with special powers whose skill level exceeds what can be normally expected from technologists on the job market. Their expertise enables them to negotiate more freely with their employers. They can often work only one or two days a week, either for the same company or as a freelancer, and get enough cash to get by – dedicating the rest of their lives to hobbies like the hackerspace.

Finally, at the other end of the spectrum there are members or even just subscribers to the mailing list who almost never come to the hackerspace. They usually have a busy job and a loving family, and/or live far from the space, but nevertheless they can make significant contributions to life there. Even though they don't have the time to attend, they find it important to have a hackerspace in their town. Therefore, they help with money and mobilising their networks. When the space in Budapest had to move to another location suddenly, the three-month deposit and the one-year fee of the Internet connection was contributed by two such individuals – something that was beyond the economic reach of the space at the time. Most often, however, these members point relevant people and valuable equipment in the direction of the hackerspace. Hardware donations often come from them. Even though they rarely visit the actual space, they can often chime in with with high quality contributions on the mailing list or chat room.

7.6 Gentleman's Clubs and Working Men's Club

The discussion of hackerspaces as clubs is inserted at this point to point out the importance of shared discourse which makes them a scene of an engineering culture. Obviously, this is not a historical but a socio-functional element, which is why this section is not part of the historical recuperation conducted earlier on, but these later sections positioning hackerspaces in the mechanics of society at large. I argue that even a cursory look specifically at Gentleman's Clubs highlights under-appreciated aspects of hackerspaces.

[90] Something that was much more difficult to do with the independent security researcher community then with the hackerspaces folks, for instance.

Although the expression *hacker club* is seldom used by practitioners (outside of Brazil where notable examples are the Garoa Hacker Clube in São Paulo and the mobile Raul Hacker Club in Salvador), I believe that the formation has reminiscences and affinities with the classic gentlemen's and working men's clubs, especially ones organised around trades. These reminiscences definitely include a dose of misogyny and racism, sometimes resulting in at least women and transgender participants to set up their own spaces, (Toupin 2014) similar to what happened with the University Women's Club (originally called University Club for Ladies). Interestingly, Oldenburg (1989) also highlights this problem in his book and dedicates a chapter for its examination. Here it is enough to state clearly that classic clubs as much as hackerspaces appear to be sites for the performance of a particular kind of masculinity or even homosociality.

As Putman also observes (381-389), most clubs and similar associations lived their heyday during the turn of the century (in the period between 1880 and 1920). Working men's clubs are a later development which I will not focus on for the moment, but gentlemen's clubs numbered around 540 in London in 1909. Their growth was driven by the extension of franchise, the greater distribution of wealth and the formation of suburbs which offered less possibilities for social life. The increasing number of gentlemen and men of leisure sought out clubs because they offered a protected environment for some immoral pursuits like gambling, erotica and alcohol, while at the same time (paradoxically) a sense of distinction (social status) and social capital (connections). Even though bylaws usually warned against the discussion of professional and political matters, clubs were an important element in the organisation of civil society. Clubs gathered the members of the upper middle class, intelligentsia and entertainers, although such distinctions were gradually eroding. The criteria for membership aimed at bringing together gentlemen based on nationality (Scottish clubs), occupation (literary clubs), hobbies (sports clubs), and education (university clubs). Putman's survey of clubs and similar associations, as well as Oldenburg's advocacy of third places echoes Habermas' similar argument **???** about coffee houses that served as sites for the development of rational and critical discourse – a key moment in the development of the public sphere and the weight of public opinion.

I find this important because it highlights and shades the social function of hackerspaces and their significance in the fabric of late capitalist urban life, as well as the potentialities it offers for political organisation around a specific interest such as technology. The fact that hackerspaces harbour so much discussion apart from actual engineering practices is important in its own right, especially for the formation of a social group which has its own norms and shared values. Of course such cultural circuits have to be organised transnationally if they were to be effective.

Echoing the historical story of the Ur-hackerspaces inspiring others through tours and talks, Putman notes that "Lateral learning was common in the diffusion of the Progressives' ideas for increasing civic engagement. Initiatives born in one part of the country were picked up and developed in other communities from whence they spread further." (397) Similarly to the efforts of Ohlig and Weiler (2007) to create the Hackerspace Design Patterns, which responded to but mainly aided the proliferation of hacker club, rules of 19th century societies also had to be systematised due to their growth: "A so-called club movement swept across the land in the late nineteenth century, emphasising self-help and amateurism. In 1876 Henry Martyn Robert published Robert's Rules of Order to bring order to the mushrooming anarchy of club and committee meetings." (383) Common rules encouraged compatibility between organisations and helped members to recognise each other as peers.

Each Gentleman's Club have reciprocal agreements with Clubs in other cities which allows visiting members to take advantage of the services offered by another club. For instance members of the Eccentric Club (founded in 1781) in London can use the facilities of the Stephen's Green Hibernian Club (founded in 1840) in Dublin free of charge. The latter claims to have 200 reciprocal clubs all over the world and some of those agreements are more than a century old. In a similar vein, each hackerspace welcomes members of other hackerspaces when they are visiting the city, and the visitor can often rely on the hospitality of local members. However, in the case of the hackerspaces, reciprocity also works through telecommunications networks that offload agency from members once they have implemented the necessary systems. The second case study (on *door systems*) outlines the mechanisms to share access credentials between hackerspaces, including physical access, network access and other logins to shared infrastructure such as wikis for documentation. Interestingly, the Eccentric Club's motto "Nil nisi bonum!" (*Nothing but good*) – reminding members of their commitment to mutual cordiality – closely resembles the motto of Noisebridge. Another famous Gentlemen's Club is referenced in a presentation by Saumil Shah about his dozen year experience in the hacker scene, at hacker conferences and at hackerspaces **???**. As a practitioner, he also compared hackerspaces to Gentleman's Clubs:

> There are many men in London, who, some from shyness, some from mysanthrophy, have no wish for the company of their fellows. Yet they are not averse to comfortable chairs and the latest periodicals (Shah slide 58).

The original quote from Doyle goes on to say that:

> It is for the convenience of these that the Diogenes Club was started, and it now contains the most unsociable and unclubable men in town (1894).

Similarly to the real life Eccentric Club, the fictitious Diogenes Club is organised around the idea of people who are either rejecting society or who are rejected *by* society banding together to a kind of Leage of Extraordinary Gentleman which fits their unusual social needs. In such a light hackerspaces can be characterised as social clubs for anti-social people. Hackers often refer to each other and their community as “a bunch of freaks”, which is an ironic comment on media representations of the hacker stereotype. While the descriptions of the classic clubs seem overly elitist, in fact behind their image of exclusivity both the Eccentric Club and the Diogenes Club were relatively relaxed about membership, condering that other clubs were only for the Scottish, or the members of the East India Company, or for people invested in the in theatre, etc. Therefore, compared to the norms of the Elisabethian age, the brandishing manifestoes of the former two Clubs can be read as statements of universal aspirations through the inclusion of transgressors (see the discussion of the Ur-doorsystem in Section 10.2.1 on universal aspirations built into unfinished artefacts). Nonetheless, both old and new social clubs have very real standards of membership, written and unwritten prescriptions about the expected behaviours and backgrounds of members and visitors, their right and responsibilities, and so on and so forth. Then and now, these are much more explicitly defined and closely circumscribed than in stereotypical institutions of the liberal public sphere, such as the coffee house or the comments sections of websites on the Internet.

As I argued in the previous section, a sizeable section of hackerspace members 1come from the most privileged section of workers (urban, educated, white, male, ideologically overpaid, etc.), and by definition all of them have the “leisure” to spend time in a club in addition, or instead of family and work. These characteristics tie them in with the gentlemen clientele of the clubs of traditional clubs, as well as the criteria of their technical skills – “a good education” was often a proxy for a gentlemanly statute in Victorian England. On the other hand, the rise of hackerspaces can be attributed to very contemporary concerns around the post-Fordist restructuration of the economy: precarity (the lack of a workplace or in certain periods even work), flexible working hours (making it harder to negotiate a regular social life), the project order (eternally changing configurations of collaborators), the increased importance of social capital (necessity to have connections outside of actual working relations), and the centrality of ICTs (the need to keep up to date with technology trends) are challenges which hackerspaces address. Yet, where they come together is their role in the development, exploration and cultivation of the pleasures, attitudes, and culture of a particular social group.

Joining a hackerspace is a cultivation of an identity: members became hackers through their activities in the hackerspace in the way that gentlemen exercise and produce their gentlemenness through participating in gentlemen’s clubs. The hackerspace is the site to discuss, explore, promote and cultivate an ethos of technology in the broad context of an engineering culture which is sensitive to the social context of artefacts. Writing about the shared value system of fraternal organisations, David T. Benito observes that “By joining a lodge, an initiate adopted, at least implicitly, a set of values.” (2000, 27)

However, the social function of hackerspaces is perhaps diametrically opposed to the social function of clubs around the turn of the century. Buchanan (1983) traces the social history of engineers acquiring respectability as a social group and significance as a profession. They choose to do it through establishing a wave of societies modelled first after gentlemen's clubs and then increasingly on the Royal Society (Institution of Civil Engineers in 1818, Society of Engineers in 1824, Institution of Mechanical Engineers in 1847, and then many more). Members rose sharply from 220 in 1830 in a single society to 38140 in fifteen societies in 1910. Buchanan muses that political quietism was the price they paid for the entry the realm of gentlemen, since they seem to have kept a measured distance from both ideological debates and day-to-day parliamentary politics. As far as we can compare these "learned societies" to hackerspaces, they were clubs uniting the social life of a profession using a formal organisation to harbour an informal milieu. That was a particular paradox of all clubs, and especially the societies of engineering which were in a transitory moment. At this point no official licence, university degree or sanctioning organisation (such as the Bar for lawyers) existed in the UK – the learned societies were a step towards the institutionalisation of engineering as a profession and engineers as an organised social group.

Hackerspaces are rolling back this process of institutionalisation of the engineering profession. By now engineering is a venerable profession held in high esteem, with a developed network of university departments and professional organisations. It is well known that the hacker scene is questioning institutional tituli, favours demonstrated skills over papers, autodidactism over formal education and amateurism over credentialism: in short, practice over theory. Therefore, it is arguably harking back to the time when engineering was a practice rather than a profession, and engineers were craftsmen in quest for the reputation of gentlemen. Needless to say, the result is a structural similarity in the social content and dynamics of the scene, but an opposition in its social function. Both hackerspaces and learned societies aim(ed) to produce a social basis for a largely amateur practice organised informally, but the former went for professionalisation while the latter go for de-professionalisation.

Having surveyed the overall decline in civic associations in the course of the twentieth century, Putman calls on the establishment of organisations that answer to the needs of members while instilling a sense of "civic engagement" in them. My argument is that such a "civic engagement" – establishing ties with people outside of the world of work and family, as well as acquiring professional standards that neither instilled – is a principal function of hackerspaces:

> What we need is not civic broccoli – good for you but unappealing – but an updated version of Scouting's ingenious combination of values and fun. I challenge those who came of age in the civically dispiriting last decade of the twentieth century to invent powerful and enticing ways of increasing civic engagement among their younger brothers

> and sisters who will come of age in the first decade of the twenty-first century (406).

7.7 Relative autonomy of an "industry relevant hobby"

Bringing together various arguments from the sections above, here I argue for the *relative autonomy* of hacking culture. Relative autonomy is the answer to the question of why critical engineering practices are possible at all? How come that given the resources of hegemonic states, the wealth of corporate giants and a largely admired academia, hackers can act and think differently? The short answer is that they cannot: by and large their culture and practices, technologies and infrastructures are penetrated through and through by these larger social structures and the norm that those imply. However, the long answer has many interesting nuances that show hackers contest hegemonic concepts of technology, organisation, and society, while sometimes they actually have initiative and go on the offensive. Whether hackers shape technology or hegemonic ideas of technologies shape hackers, though, it is only through and only in conjunction with some allied states, companies and academic actors that they can do so.

The highest profile cases of the larger hacker scene perfectly illustrate such dynamics: WikiLeaks mastermind Assange had to rely on the state of Venezuela to protect him for a while, and next year Snowden relied on Russia. The role of capital in FLOSS development is also well known: Weber (2004) argues that the history of FLOSS is the history of corporations. The hegemony of hackers' beloved UNIX system architecture has been established recently by Apple (with OS X and iOS) and Google (with Android). I illustrated the partial role hackerspaces can play in innovation networks in the section on *Hackerspaces as a black box* . At the same time these historical processes also illustrate how the hacker culture profoundly influenced the development of hegemonic technologies since the advent of the computer.

I use the term relative autonomy to capture this ambiguity. Relative autonomy allows for three things. First, hackers can have their own critical opinions and imaginaries which can inform critical engineering practices. Second, hackers can realise their techno-political ideals through running their own infrastructure and producing their own artefacts as the social and technological basis of an engineering culture. Third, hackers can participate as a stakeholder in techno-politics alongside the state, capital and the academia as a loud minority of ICT developers, maintainers and users.

It is important to realise how the hacklab model around the turn of the millennium allowed for greater autonomy – defined as independence from hegemonic currents and the ability to produce one's own culture, practices, and technology. However, such autonomy resulted in isolation to the activist ghetto, given less opportunities for intervention and less resources for the development of alternatives. I argued that the hackerspaces model seems to strike a balance between integration/interaction with larger social structures and the maintenance of autonomy — albeit at the cost of loosing a consistent political perspective.

I claim that relative autonomy rests on three pillars. First, expertise: the hackers' ability to understand and produce technology. Second, shared values and culture traditionally transmitted. Third, a historical horizon based on the familiarity with older technologies. These three are all necessary conditions and together they are sufficient for adopting a critical point of view over hegemonic developments of technology. In the remainder of this section I demonstrate why one pillar would not hold without the others, then illustrate how each pillar is produced and reproduced in the lived experience hackers, and finally cite three observations to show semi-autonomy at work vis-a-vis the state and capital.

Carrier engineers have three steps in front of them: first, they typically begin as hobbyists in their teens, then go to university to study informatics (physics, etc.), and finally get a job in a tech company or perhaps start their own. Even computer scientists can walk this path without ever getting in contact with the engineering subculture of hackerdom. Such an education gives them enough expertise to be able to understand and produce technology (the first pillar). However, since they are only familiar with the hegemonic interpretation of technology, it is hard for them to arrive at critical conclusions about it simply because as most culture, the mainstream engineering culture is largely self-serving and self-reproducing. Those who do may eventually join the hacker scene.

Given both engineering skills and hacker attitude it is still hard to question the contemporary developments one grows into if a historical perspective is lacking as a ground for comparison. This second step in my argument is admittedly more contentious: after all, how come that its history is not part of a culture? How can one acquire a culture without getting familiar with the associated historical experiences? These are valid questions to ask. But then, Lindtner and Li (2012) shows that Asian domestication of hacker culture could do away with its political values, arguably because a lack of historical experiences and context. Many finer, tacit points of hacker culture rest on generational experiences that include immersion in specific technological and organisational regimes, and these need specific channels of transmission.

Hackers' ability to understand and produce technology is important in a number of ways. Understanding how something works is a prerequisite to judging it, and being able to improve it or make something better is empowering because critique can be constructive. Much of the confidence that hackers display in the political arena when compared to mainstream activist discourse rests on such contributory expertise. Activist practices such as "protest" are generally reactive, and even "prefigurative politics" — alternative practices which defy hegemonic ways of doing things – have a hard time to influence how things are done elsewhere. In contrast, hacker's relative autonomy allows interventions through the creation of artefacts (software or hardware), which can alter how things are done in the industry, or change technological possibilities that in turn can influence policy making. Of course hackerspaces help members to acquire, improve and exercise the skills necessary for understanding and producing technology. This constitutes a breakdown in the experts versus amateurs debate since outsiders can intervene with technical arguments and prototypes. Expertise wielded from the grassroots challenges the managerial role that is usually fought for or offered to citizens in political debates about technology. The difference is that a managerial role means making high level decisions about development that one does not fully grasp, whereas hackers can criticise, improve or obstruct technology from the inside.

I hope to have showed to some extent how shared semi-autonomous culture and values are produced and reproduced in hackerspaces, which provide a materially and symbolically constituted milieu. A hackerspace is a kind of shrine to hacker culture which seems to encompass everything from culinary preferences through technical adages to religious references. Thus an engineering culture is comprised of more than technical matters, and technical aspects of the culture are reinforced by the references to other spheres of lives. The fridge mainly stocks Club Mate, the default drink of hackers; (Thomas 2014) the soldering iron stall warns that "If it smells like chicken you're holding it wrong"[91], a reference to Mitch Altman's soldering workshops; and names of people and machines are taken from Discordianism,[92] the Cthulhu mythos,[93], or The Hitchhiker's' Guide to the Galaxy.[94] Thus a specific taste, practice and metaphysics is upheld that completes and legitimises the technical teachings at the heart of hacker culture. Rather than analysing the contents of these further – since that is the mission of the case studies – the point advanced here is simply that hackerspaces display a richly marked cultural and ethical difference, as the most complete and permanent manifestation of hackerdom that is available as lived experience.

In the previous section I surveyed hackerspace membership and how they contribute to the political economy of the hackerspace. One conclusion was that the economic sustainability of hackerspaces depends on a core number of members who have ideologically overpaid jobs in the technology sector, as well as on associated companies which feel compelled to support hackerspaces in a number of ways. Working for these companies which often promote a mainstream engineering culture themselves becomes the material basis for running other organisations that promote an alternative. Kelty (2013) notes the relative affluence but still hard and often boring work of many hackers, writing that "many software developers [are], toiling as they do in the richer veins of freelance precarity." Hackers

[91] A joke on grabbing the hot end of the soldering iron.

[92] Popularised through the Illuminatus! Trilogy by Robert Shea and Robert Anton Wilson.

[93] Mainly associated with the horror writer H.P. Lovecraft.

[94] Written by Douglas Adams.

as in some sense the most privileged category of workers (white male highly educated inner city dwellers, etc.) use these resources to establish trends which are at least disturbing the hegemonic concept of technology. Even if these are often and quickly recuperated into the mainstream, the same process establishes a channel of interaction that goes two ways, allowing hackers to participate in steering technology and shaping the orders of worth associated with it.

Historical familiarity with previous technologies is passed on largely through a floating debris of obsolete gear which is a permanent fixture in the landscape of hackerspaces. Some go further than others. Perhaps the most symptomatic case is Hack42 in Arnhem, the Netherlands. Housed in an old military barracks modelled after German countryside cottages, the three storey hackerspace includes several thematic collections of obsolete hardware.[95] Firstly, cameras analogue and digital, including associated gear such as overhead projectors and beamers. Secondly, typewriters which are in use regularly during typewriting days. Thirdly, calculators and computers many of which are also in working order. As much as anything else, the hackers' idea of a museum is based on the hands-on imperative: practice over representation. The Museum of Functioning Informatics located in the UNESCO heritage site Palazzolo Acreide (Sicily) methodologically promotes a hands-on approach to museology. It is a side project of FreakNet hacklab, which itself is maintained to be the oldest of its kind.

Hackers often appreciate the efficiency, simplicity and transparency of older machines which were in many respects easier to use, understand and modify. On the other hand, when probed about the relative advantages of modern machines, it is customary to cite the mathematical theory of *Turing completeness* which circumscribes the realm of logical operations a computer in general can do. Since all computers are Turing complete, in a logical sense there is no difference between their capabilities. Of course as I argued in the beginning of this chapter, recycling is the backbone of the political economy of hacklabs, and in a more pedestrian sense every shared machine shop needs its own junkyard as a source of spare parts, fixable fixed capital and last but not least: inspiration. Old machines, unfinished projects, and complete devices which are stored on the shelves of hackerspaces serve as a cookbook of engineering solutions. Asking for the proper way to wire a chip, one can find such chip already wired into some other device and use it as an example.

Repairing, using, and developing these machines can be considered critical practices which question the modern narrative of technological and therefore social progress. However, a more safe claim is that nostalgic appreciation of old machines (as much as reading science fiction) at least sensitivises practitioners to alternative paths that technology could have taken, which is arguably an advantage when it comes to judging current developments. As I show in Maxigas (2015b), hackers are not merely developers or early adopters of new technologies: they are sometimes significant saboteurs or stubborn non-adopters. A case in point is the low adoption and frequent critique of smartphones as surveillance

[95] A partial list is available here: https://hack42.nl/wiki/Hardware_lijst_controle

devices based on proprietary communication protocols that prevent users from making use of their full potential. Perhaps the most lucid example of the scepticism over the direction ICTs take is Appelbaum (2011) who forcefully opposes the trend of tablets, ebook readers and smartwatches, calling hackers to prepare for The Coming War on General Computation.

A more material case is Mitch Altman's signature invention, TV-B-Gone:[96] the universal remote control with a single button that can turn off any television. While the former is a theoretical argument about an all-encompassing historical trend, the latter is a practical device that can be called the material residue of critique. At the same time Doctorow's critique attacks emerging technology trends but Altman's device attacks the past that persists.

Hackers wielding the TV-B-Gone express the anti-television sentiments of a whole generation of computer users. They ditch a large and lucrative segment of the electronics industry which they deem degenerate, bent on a quest for half-solutions and vaporware such as set-top-boxes, smart TVs, 3D vision, ever wider screens and the mythical "convergence" which would make the Internet a television. Such sabotage is in stark contrast with the the long struggles of hackers to bring about personal computers in the 1970s, described at length by Levy (1984). I argued that fighting for certain innovations while refusing others necessitates familiarity with subsequent generations of technologies, an experience that hackerspaces support through stocking old hardware which members can use or take apart.

Thus, autonomy is relative because it does not achieve or aim at complete independence and self-sufficiency, or one could say sovereignty, from the state. This is in stark contrast with hacklabs, which usually operate without a legal body and inhabit autonomous zones of some kind. So while hacklab members can hide effectively behind pseudonymous monikers without further questioning, hackerspace members can call each others names, but in most countries they have to give their real name and address to become members. As anarchist initiatives, hacklabs oppose the state on an ideological basis and practically confront it heads-on at least through occupying property. In contrast, hackerspaces question state legitimacy playfully. I bring two instances of this practice and then briefly analyse their ideological import.

The hackerspace passport[97] is document where visitors to hackerspaces can collect stamps called "visas". It physically imitates the look of a passport and features the lettering "DIPLOMATIC PASSPORT" on the cover, a special page for personal data and photo identification, as well as a number of visa pages with hackerspace logos watermarked into the background. Just as some states include

[96] See https://www.tvbgone.com/

[97] See https://www.noisebridge.net/wiki/passport

parts of the constitution in their passports, this one has a manifesto starting "There is a world of hackdom out there."[98] Many hackerspaces and some other organisations make their own stamps which visitors can collect in their passports. Thus in practice the hackerspace passport is not an identification document per se, neither it functions as an instrument of access control. It imitates booklets where scouts and other people touring an area can collect stamps of sites as memorabilia, similar to the little flags hanging behind the windscreens of trucks, national flags sewn on backpackers' luggage or stickers on trunks. However, it does symbolically identify the holder as a member of the hackerspaces scene and practically helps to gain recognition in hackerspaces. It has been created by Mitch Altman following the suggestion of Egerlach on the hackerspaces wiki, and designed by Matthew Borgatti. The hackerspace passport is widely recognised in the scene and when hackerspace members come together in hacker conventions, it is possible to gather several dozen stamps in a single day. I used it myself to gain rapport during field work and ordered a reprint in a local shop to distribute blank copies to hackerspaces running out of it, and got out of a few airports using it as identification.

Since issuing passports is a principal privilege of sovereign states it is a practical joke calling into question their legitimacy, as well as the performative powers of any bureaucratic document that is supposed to be more than a specially designed piece of paper. Such practice fits the pattern outlined in the black box section earlier, where hackers take a technology from the military-industrial-academic complex and make a punk version that is cheaper, easier to understand and reproduce, and endowed with a different set of imaginaries. At the same time it addresses a key obsession of hackers, an area where the successful performance of engineering expertise is crucial: security. While the Hackerspace Passport can be categorised as "parody counterfeit", the Chaos Communication Congresses of the last years featured booths where one could make valid German ID cards as part of the campaign by the Chaos Computer Club criticising the security level of the built-in biometric features. Here again, the primacy of the successful performance of sovereignty by potentially "rouge actors" over the poor performance of sovereignty by those officially entitled is thematised. The gesture thus echoes themes of technical expertise wielded from the grassroots versus managerial control by citizens in political controversies, as well as the primacy of skills over credentials for the recognition of good engineers. The most important message of the hackerspace passport is that hackerspaces are an supranational movement not tied to borders and border controls, with the tacti assumption that technical expertise in particular and learning in general cannot be constrained by state actors.

The Hackerspaces Global Space Program launched in 2012 during the Chaos Communication Camp with the mockup goal to "send a hacker to the moon in 23 years". While 23 sounds like a very precise date, it is in fact a reference to

[98] The manifesto is not analysed in detail for several reasons, but mainly because it is probably too tied to the American context to be widely read and appreciated by European hackers.

the aforementioned Illuminatus! Trilogy where it is a magic number that the characters find everywhere in the world around them, which either proves the veracity of Discordian metaphysical teachings or the hegemony of the Illuminati conspiracy. The programme provided a framework for hackers to discuss (many) and implement (some) space-related projects such as autonomous satellites, stratospheric photography using meteorological balloons, building open source equipment for space stations, and so on. It served as the main theme of the Camp in 2012, developing on the mascot of the Club, an iconic spaceship statue called Fairydust which appears at all their events. In fact a network of mostly US hackerspaces actually wrote up an application to DARPA (the main body for military research funding in the USA) which included similar ideas. However, for most hackers it was just part of a practical joke for that summer making megalomaniac claims about the power of the scene and making fun of nation states who gain recognition by putting people on the moon, as well as the megalomania for their emulation by emergent private space agencies like SpaceX. Since the last German hacker camp took place in 2008 – is is organised in four year cycles — it was also a recognition that hacker *spaces* experienced and exorbitant growth in the last years and such a wide spread infrastructure of clubs staffed with such awesome people could accomplish anything they could imagine. In fact the growth of peer production projects is notoriously hard to predict: an idea like Wikipedia may never work in theory but it works remarkably well in practice.

Space programs are a characteristic targets for hackerspaces because for three reasons. Once, they are prime examples of using scientific research and engineering achievements with larger-than-life budgets as public relations instruments that bring prestige to their funders and founders. Twice, in the same mythical capacity they are the ultimate win in the arms Cold War arms race between the USA and the USSR, underpinning the legitimacy of the whole enterprise of these states. Thrice, after a period of *space winter* of low funding and enthusiasm from the 1970s on they have been reactivated to fuel the myth of Internet monopolies, the chief example being Amazon CEO Jeff Bezos' Blue Origin which is competing against PayPal and Tesla Motors co-founder Elon Musks' SpaceX. In fact they are as close as state and capital actors ever got to the hacker justification for doing the amazing "just because I can". Therefore, launching the space program – tongue in cheek as well as an actual bid – has been the next logical step for hackerspaces in the road to world domination. Characteristically, the only related hack during the actual camp was sending a radio transmission which bounced off the moon and was received on the other side of the planet. Earth-Moon-Earth communication is no doubt virtuoso trick of radio operators since it uses the celestial body as a passive satellite. However, the technique is hardly new because its first application was to link Pearl Harbour to the military command centre in Washington during the Second World War and the achievement has been replicated by radio amateurs since the 1950s. Thus, even the moon bounce in itself can be interpreted as a critical gesture that displays the pillars identified so far, e.g. technical expertise, shared culture, and historical awareness.

In short, hackers are bothered by state sovereignty for two diametrically opposed reasons. Firstly, state sovereignty interferes with what they see – from a very liberal, even libertarian political point of view – as the "natural rights" of the individual. Secondly, as the builders of infrastructures and communities (which they claim pay more respect to the rights of individuals), they see themselves in a sort of competition with state sovereignty. Relative autonomy offers a platform that can be used to question state sovereignty as far as it remains a symbolic gesture merely addressing its legitimacy.

In the larger scheme of things these gestures are typical gestures of cosmopolitanism which posits that all sentient beings belong to the same community (a profoundly democratic notion), but also posits that for the same reason the ideals of cosmopolitans apply to all (a clearly authoritarian notion). As far as hackers stand for the interest of technology users in general, what is good for the hackers should be good for everybody. Such avantgarde position is justified by their relative autonomy: their expertise that allows them to know better, their semi-independent position from the organs of the state and capital, and their "long duré" view of the historical moment. Of course, the very definition of social privilege is that it can effortlessly appear as the universal, and we have already commented on the position of hackers as in some ways the cream of the working class. Fortunately, it seems that in the present historical moment the partial interests of hackers as a particular social group coincide in many important respects with the interests of many more populous and less privileges social groups.

The continuity between these contemporary manifestations of hacker culture with the scientific imaginaries of 1960s counterculture which is described by Markoff (2005) and Turner (2006) is easily recognisable in the configuration of this blend of cosmopolitanism. Turner identifies Stuart Brand's Whole Earth Catalogue as a site of emergence for such vision, reminding the reader that the title of the catalogue came from the famous first photo of the Earth from orbit. The hackers' world citizenship and intergalactic projects resonate with this imaginary. The idea of exploring unknown territories is often summoned in other projects as well, such as the title of the first major hacker meeting in the Netherlands – Galactic Hacker Party; the slogan of one Chaos Communication Congress – Hic Sunt Dracones;[99] or the name of the hackerspace in Amsterdam: Technologia Incognita. We will see later on how an ontology, ethics and techniques of the unknown is developed on such basis. Thus is will be possible to understand both the formal basis for collaboration, both the essential content of hackerdom. For now it is sufficient to see that relative autonomy unfolds in a cosmopolitan outlook that encompasses the whole universe, and it is used to exercise undirected creativity in such an environment. Hackers situate themselves in the universe, not within the private or public sectors, and they see the universe as full of

[99] A reference to the traditional legend on medieval maps to mark unexplored territories.

mysteries to be discovered, half revealed, half obscured. Thus, it can be argued that in contrast to the hacklab strategy of total autonomy based on isolation from the state and capital, relative autonomy aims to transcend them.

To summarise, I argued that hackerspaces have to be understood in the context of the relative autonomy of the hacker scene. The three pillars of relative autonomy are technical expertise, shared values and culture, and a historical horizon. They allow hackers to cultivate critical opinions and engineering practices, reproduce their social basis through maintaining hackerspaces and intervene in the techno-political arena about the direction technological development takes. Relative autonomy is a term that seeks to capture the fact that the political economy of hackerdom is intimately intertwined with those of the state and capital, but paradoxically allows a circumscribed area of cultural autonomy that includes engineering practices which are sometimes construed as interventions into larger techno-political processes.

7.8 Conclusion: Unfinished Architectures

The ensemble of characteristics described above define what I would call *unfinished architectures* in the latter part of the dissertation. Unfinished architectures provide an organisational framework for open-ended collaboration that allows for undirected creativity and unalienated labour. In such an environment where life and labour are not divided from each other, there seems to be at the same time an infinite sloth – discussions and socialisation – and an infinite productivity – hacking and engineering. The informal sociality of peer production allow the productivity of life as such to be developed without compartmentalisation. When the market is not the primary determinant of social behaviour, the methodological individualism of economics breaks down. This explains the difficulties of transaction cost theory introduced in the theoretical framework to account for peer production phenomena. Unfinished architectures can give reign to a wide range of energies which are not captured by the more powerful determinants of the modern institutional grid.

Hackerspaces are a specific achievement of social innovation that has been developed in the confines of the hacker scene, building on previous processes like the constitution of a specific engineering culture, partial institutionalisation of hacking in negotiation with larger social structures like the authorities, media and the state, and the relative autonomy of hackers as a social group that allows interventions in social, technical and political matters. It can be argued that hackers hacked hackerspaces together in order to facilitate peer production practices to be developed outside of the confines of the modern institutional grid, as a niche protected from pressures associated with the market, the state and to some extent with civil society. This is what makes them a privileged site for the study of peer production processes, especially hardware which is necessarily developed through haptic collaboration.

It is clear that unfinished architectures leave open many possibilities for finishing them, e.g. to orchestrate the whole organisational dynamics to serve a single set of well-circumscribed ends, be that the design of new products, project based learning or fostering forms of sociality that lend themselves to the development of innovation networks. In the closing section of the chapter which serves as a kind of appendix, I show how all these are done in other genres of shared machine shops. The traditional trajectory of hacker culture keeps hackerspaces within the limits of unfinished architectures in the face of recuperative possibilities.

8 Shared machine shops compared to hackerspaces

The objective of this section is to provide a glossary of the rapidly proliferating genres of shared machine shops, as a sort of appendix to the chapter. These are interesting for three reasons. First, since most of these categories have been established after the rise of hackerspaces, they bear testimony to the impact of hackerspaces outside of the scene. Second, they allow the reader to contextualise the research in the related literature which is mostly about these other genres. Third, references to these next waves of shared machine shops will crop up in the next chapters.

I argued that a big enough fraction of the participants has well-paying and more or less stable jobs in the technology sector, which ensures the viability of the membership-founded model, but individual and corporate donations are still an important factor in the development of the hackerspace. These are complemented by the more complicated role of hackerspaces as the “authentic” scene – a position only strengthened by the arrival of more commercial Share Machine Shops like Fab Labs, Makerspaces, etc. Authenticity here is used in the sense of Boltanski and Chiapello (2005) meaning something which is produced without secondary intentions and something which is not aimed at market circulation. Despite this definition – as the latter authors point out – authenticity became an integral part of contemporary patterns of capital accumulation. This is demonstrated in detail by Fleming (2009) or Liu (2004). I investigate the dynamics between the movements of technological capital and the self-organisation of technology workers in Maxigas (2015b) in the framework of recuperation set out by Söderberg and Delfanti (2015).

I have demonstrated what I mean by hacking being “an industry relevant hobby”. While hackerspaces are often interpreted as *innovation spaces* or similar in the literature, it has to be emphasised that they can be but one node in an innovation network. This is due to the fact that as one self-satisfied maker put it grudgingly, “hackers never finish anything” (again, thanks for Debora Lanzeni for this data) – a reminder that most of the innovation process happens beyond prototyping. Prototyping is only a small portion of the trajectory from design to valorisation on the market. Nonetheless, hackerspaces do play a significant role in the process, similar to underground music clubs where experimental bands play and once in a while managers recognise the potentials of marketable talents: prototype bands. In fact many companies, universities and incubation spaces mimic the hackerspaces model by establishing their own similar facilities in-house. The rationale for these attempts is that the free experimentation which is the distinguishing mark of hackerspaces increases the value of the employees, students or entrepreneurs working in the establishment, and leads to new ideas, experiences and prototypes which can be potentially turned into products, academic research or serve the kernel of technology startups.

Previously I have argued that undirected creativity and unalienated labour are at the centre of the hackerspaces ethos, and hackerspaces fall out of the modern institutional grid, so that now I can distinguish them from other genres of shared machine shops based on these criteria. In fact subsequent genres can be analysed as so many "rip-offs", e.g. more or less successful attempts by other actors to direct toward their own goals the undirected creativity inherent in the hackerspaces model, or exploit the unalienated labour enabled by the self-management of hacker clubs. The order in which genres are discussed below loosely follows three criteria: time of emergence from earlier to later; degree of recuperation; sociological distance from the hackerspace model. Of course the simple possibility of listing them thus carries the claim that these three factors are correlated, but I will not make a sustained effort to prove that hypothesis because it would derail the focus on hackerspaces themselves. These sections are loosely based on similar surveys such as Cavalcanti (2014) or Cruickshank (2014) 69-73.

8.1 Makerspaces

Makerspaces are hardly distinguishable from hackerspaces, the main difference being symbolic and discursive (J. aka k. Moilanen 2012a). They have renounced the hacker moniker and thus the solidarity with the hacker scene. The hacker scene is organised around some idea of workers' solidarity, e.g. research and experimentation with technology supersedes division lines which are establishes by other social factors. Black hat hackers – in other words criminals – can have discussions with activists – sometimes termed terrorists by authorities – and law enforcement agents – often called pigs by anarchists. Independent researchers (the grey hat hackers) can discuss with those who work for states or corporations to secure their networks and academic scholars that specialise in cryptography. While mixing these social groups can lead to conflicts, hacker culture provides protocols to negotiate these tensions. Whatever the internal dynamics of the scene is, the image of hackers in the eyes of the general population are marked by both fear and fascination. Makerspaces renounce both the social practices and the imaginaries of hackerdom in order to present a smoother image to the public. Internally there is also a shift away from the traditional hacker relation to technology – sometimes formulated as "making and breaking" – which concentrates on the clever tricks, surprising wits of engineering bravado, and the neat solutions. Makers see themselves mainly concerned with creative expression through technical craftsmanship which does not have to involve the exploration of the limits of systems. Therefore in some ways their endeavours are less ambitious and inclusive internally as an engineering culture, but more ambitious and inclusive externally as outreach. While some makerspaces would double-bill themselves as hackerspaces, or some makers would tell you privately that in fact they see their work as hacking but prefer not to frame it as such discursively, many other makerspaces and individual makers renounce hacking altogether.

Having said that, the atmosphere, the social organisation and social relations, the political economics and the technical projects of a makerspace are more or less the same as a hackerspace. In fact the phenomena of makerspaces can be interpreted as the American reception of the hackerspaces ethos, implanted in a cultural context where hacking still counts as controversial, and despite its institutionalisation the hacker scene have not managed to acquire as much acceptance and respect as in the old continent. On the other hand, the makerspace label allowed these organisations to integrate into a rich and booming DIY/DIWO[100] craftsman tradition in the United States. A third factor is the specific process of institutionalisation and cultural transmission which happens to a much larger extent through commercial mediation. The role of Maker Media in the establishment of the makerspace scene in the United States cannot be overestimated. Maker Media have been associated with O'Reilly Media, the de facto standard publisher of technical manuals for FLOSS languages, frameworks and applications. Maker Media's MAKER magazine is as important a source for North American makers as the online blog Hack-a-day in Europe: a source of ideas and documentation, but also role models and cultural traits. Maker Media at some point won a major grant from DARPA for the education of secondary school children. Part of the project implementation was to register makerspaces.com and brand it according to their idea of makerspaces as civic communities. The decision was not really welcome even amongst makerspaces and the project largely faultered. Critiques compared the community-run hackerspaces.com website that maintains a self-managed directory of hackerspaces to the corporate groomed collection of makerspaces unfavourably.

The global hackerspaces discussion list more or less predictably breaks out into a heated discussion around the definition of hackerspaces in general and the difference from makerspaces in particular. The highly confusing and largely inconclusive nature of these debates shows three things. First, that hackerspace participants (on the list) feel strongly about the hacker(space) identity. Second, that the distinction between hackers and makers is in fact not that great even if the fine lines of distinction are heavily contested. Third, that the hackerspace crowd feels closest to makers than to Fab Lab researchers or Tech Shop employees (more on these later) because it seems to be easy for them to differentiate themselves from participants of those more recuperated genres of share machine shops.

8.2 Fab Labs

The creation myth of Fab Labs is the stuff of well documented legend. Kohtala and Bosqué (2014) observes that "In spite of this emerging body of work, there remains a certain mystique that surrounds Fab Labs, their objectives and activities. … Surprising little has been written about the germination of the first Fab Labs aside from Gershenfeld's own account (2005)." Commentators

[100] Do It Yourself / Do It With Others.

seem to agree that Neil Gershenfeld founded the Center for Bits and Atoms at MIT in 2001. The original research question had to do with the separation of form and content, e.g. designing systems digitally that can be implemented in physical or biological mechanisms. While scientists continue to work towards this goal, the short-lived Grassroots Invention Group along with the "How to Make (Almost) Anything" class inspired the global proliferation of Fab Labs: first in spectacularly *poor* areas targeted by the MIT such as a Boston ghetto, Northern Norway, Ghana and India; (Kohtala and Bosqué 2014) then in the "usual suspect" core countries such as the Netherlands and Japan. However, the latter movement proved more spontaneous and harder to subjugate. Troxler (2015) documents the "A Grassroots Insurgency inside the Next Industrial Revolution", or the rise of self-organised Fab Labs that neither conformed to the specifications set by MIT nor willing to pay the trade mark fee. Rouge Fab Labs managed to diversify the Fab Lab landscape and bring in community involvement.

The Fab Lab concept can also be taken as an example of a largely failed recuperative attempt on the hackerspace model, but this time not from the private sector (e.g. Maker Media) but the academia (namely the MIT[101]). Interestingly, both include an attempt to capture into precise specifications some of the external properties of hackerspaces, specifications which can serve as a stable basis for expansion. Of course what is left out of the specification is the most interesting part for understanding how recuperation works. As in the previous case, the expansion was imagined as the rolling out of a franchise, but engaging with the grassroots complicated things greatly.

The equipment of Fab Labs is precisely defined down to the level of model numbers in order to foster compatibility, and therefore cooperation between the nodes of the network. In general terms these machines are meant to aid "personal digital fabrication". The former adjective has undoubtedly been chosen to suggest a future quantitative ubiquity and a coming qualitative social impact comparable to the personal computer. However, the rigorous observer has to note that the power of the personal computer have been unleashed in conjunction with its ability to mobilise cooperation in a broadly understood social network beyond "personal expression in technology". Here may lie an explanation to why the mission of the Fab Lab network — a globalised cooperation driven by digital design and communication networks between geographically dispersed community laboratories – have scarcely been realised. Wolf et al. (2015) survey the situation and conclude that collaboration between labs is at the moment rather a symbolic ideal than a daily practice.

The original idea of outreach at MIT was remarkably similar to the general impulse of hacklabs: "to empower people to become technological protagonists rather than just spectators" (Gershenfeld 2005). In other words, to exchange the managerial attitude to a technically prolific and productive one — grassroots technology research and development. Notably, such goals are in line not only with my own analysis of hacklab/hackerspace activities but the general ambition

[101] Massachusetts Institute of Technology.

of the Science and Technology Studies research program on the democratisation of expertise.

8.3 TechShops

TechShops (mentioned above) are operated as a franchise firm providing knowledge and access to machines similar to the tooling of Fab Labs: 3D printers, laser cutters, CNC machines, sewing machines, wood and metal working equipment as well as hand tools and electronics components. (Hurst 2014) The setup is similar to a photocopy shop, except that self-service is complemented by courses in using machines and community events where members can interact with each other and the staff of the store. Machine access is sold on a yearly, monthly or daily basis. Founded from donations in 2006, the first Tech Shop opened its doors in Menlo Park, CA. The next two (in Portland, OR and Raleigh-Durham, NC) were straight commercial ventures and failed in the space of a few years. Faced with these dired straits TechShops changed its strategy and started to partner with Fortune 500 corporations. In Detroit, TechShops partnered with Ford and Autodesk, while in Texas, TechShops partnered with Lowe's home improvement chain, "the second-largest home improvement retailer in the world" (as reported in the online magazine called Informed Infrastructure: The construction engineer's source for projects, products and technology, Ball (2012)). These survived, turned out profits and dozens more are being established. (Rivlin 2011)

While it is common for at least some Fab Labs to rent out machine access and organise courses for profit, TechShops can be seen as systematically deriving a business model from the way shared machine shops work. It would be interesting to compare projects that TechShop users make with projects by hackerspace members. Such a study could bring out differences and similarities in the social relations as well as the technological results between the former model which formalises and monetises interactions between people (primarily in courses) as well interaction between individuals and machines (primarily machine time is bought) and the latter where both are essentially organised on an informal basis according to the principles of peer production. Even though hackerspaces also charge for membership itself, access to people and machines is free for all who happen to drop by (as stated before). Yet, both genres of shared machine shops use a similar range of equipment and rely on almost the same range of Internet platforms (Thingiverse for 3D models, Maker Magazine for project ideas, etc.) – with the possible exception of IRC, which remains a social media almost exclusively utilised by hackers donning various colours of hats. However, such comparative data is not available for the moment, so I can only rely on published sources that are largely journalistic accounts.

TechShops are little known in Europe and I could not find similar ventures, probably because of the differences in entrepreneurial culture and practices, as well as the geographically specific reception of hackerdom. In summary, TechShops are a North American phenomena of Hacking-as-a-Service, emerging almost simultaneously with hackerspaces – an obvious example of capitalising on the hackerspaces model. While taking part in Maker Faires and repeating some of the symbols of the hacker scene, TechShops do not give members ownership and control over the organisation, the means of production, or the premises.

8.4 Accelerators

Accelerators like Indie Bio have been mentioned in conjunction with the last wave of technologies – namely synthetic biology – which have been absorbed into the hackerspaces scene. A relatively new model of capitalisation looking back to a history of less than a decade, accelerators take teams with ideas, trying to turn them into startups. Somewhat like hackerspaces, they provide a conductive environment for the development of projects that combines social relations between members and infrastructure for technological research and development. However, members are chosen by the managers of the accelerator to participate in a single "run" of a few months, coached by consultants and geared to enter the market. Since accelerators focus mainly on emerging technologies, they are perhaps the pivotal example of the tendency after the turn of the millennium for hackers to achieve a similar social position to rock stars – hand-picked, produced and promoted by recording studios.

If hackerspaces are examples of the workers' control over the means of production, or the self-management and self-organisation of ex-workers in order to exercise undirected creativity and experience unalienated technological labour – if hackerspaces are a milieu where workers' discipline is technically and culturally contested, where process-based projects are embedded into the fabric of everyday life and subcultural sociality — or privileged sites of commons-based peer production which provide access to technological knowledge and infrastructures to the public to do grassroots research and development – then the model of accelerators is diametrically opposed to hackerspaces. However, the actual success of accelerators is highly dependent on discovering ways to question prevailing notions of technology, invent practices that challenge the usual way of doing things, and enrolling a large number of volunteers in research projects – a mission closely resembling the hacker ethos promoted by hackerspaces. Moreover, both hackerspaces and accelerators use open knowledge and technologies – from software through hardware to biology — to quickly attain the state of the art and therefore being able to improve on it. Startups sponsored by accelerators typically include some aspects of peer production in their business models but complement it with proprietary components. As explained in the theoretical framework, these Benkler (2006) calls hybrid models of peer production while Weber (2004) and Tapscott and Williams (2006) presents various possible strategies for the creation of hybrid models. Depending on which requires the most material and temporal input, successful strategies open up either the core component of the business model, or exactly the opposite: supporting infrastructures and services.

Even though accelerators recuperate the hackerspaces model in many ways, from its technical repertoire to its cultural gestures, and adapt the peer production principles of organisation to direct it to the production of startups, the political economy of accelerators differs markedly from TechShops. The business model of the TechShops chain is based on selling access to knowledge and fixed capital based on discrete products metered in human work time. Accelerators are operated by venture capital firms which profit from gaining equity in startups rather than selling services to them. The fact that both TechShops and accelerators use a human work time measurement (3 months "runs" here, subscription packages there) to control access to the resources they provide is a rather formal similarity, even if it contrasts nicely with the hackerspace tradition of providing 24h hour access to members and to keep the doors open as much as possible. This perspective is further expanded in the second case study, where I show how the measurement of human work time is used in the hackerspace milieu to enable undirected peer productivity informally.

8.5 Corporate hackerspaces, innovation laboratories, media labs and hubs

Such is the cultural attractiveness of hackerspaces that even some corporations whose core business has nothing to do with digital fabrication decide to set up shared machine shops in their premises. These establishments largely serve three purposes. First, they are supposed to foster creativity and self-expression amongst the higher tiers of the work force. Second, they are considered amenities that come in addition to the salary, along with free lunch and ping pong tables, etc. Third, they allow the company to improve its public relations image through associating itself with the maker culture. However, companies like HP which is a player in the 3D printing market also use internal shared machine shops as a way to get culturally and technically familiar with the context of its source of inspiration and its target group. Additionally, in some cases shared machine shops are used to optimise or innovate the design and manufacturing practices of the firm, or to experiment with prototypes.

Since most of these shared machine shops are open for workers who are not really working at that time, they are actually closer to the hackerspaces model than TechShops, accelerators or Fab Labs. They are largely an afterthought to the social architecture of their respective corporations, and therefore can be unfinished in the sense that they are not directing creativity for a particular end. In fact their recuperative move comes on a higher level when they became part of the brand image of the company. They serve their purpose best if they are as authentic as possible until they the branding comes in.

8.6 Men’s Sheds

If the previous genres depart from the hackerspaces model to address a more privileged audience from a more commercial point of view, Men’s Sheds address a less privileged clientele. Hackerspaces balance a dual community and technology mission, while previously discussed genres enhance the technology and innovation aspect to the detriment of community and social values (even if they reproduce community themes in their public relations discourse). Men’s Sheds are explicitly framed as community organisations, and all the technology work is accommodated into that perspective. Men’s Shreds specialist Barry Golding writes that they “allow men to regularly meet and happily socialise, mainly with other men with tools, in a safe, familiar shared workshop setting” and emphasises that they “allow them to socialise, feel at home and learn informally by doing, in practical, group situations with other men.” (Ferrier 2006) Correspondingly, the hinterland of Men’s Sheds is not in central European countries nor in the United States, but in Australia. No doubt at the height of their popularity in Australia with over 600 organisations and two national networks, a numerous and growing body of Men’s Sheds appear to exist in other peripheral Anglo-Saxon countries such as Ireland and Canada.

On a more abstract level the purpose of Men’s Sheds is to provide a space for the performance of a particular kind of masculinity in a supportive and collaborative environment which is claimed to be good for older men’s health and well-being. Even though the narrative of Men’s Sheds’ health, social and educational benefits is well established in the literature mostly based on anecdotal evidence, the actual positive effects are not established quantitatively. Building on the traditional Anglo-Saxon imagery of the shed, and sometimes describing their activities as shedding, Men’s Sheds constitute a link between historically well established domestic technology practices and the more recent rise of hacking and shared machine workshops. Golding et al. (2007) conclude that “sheds in community contexts retain and incorporate some aspects of Australian masculinity, including the shed as an iconic place for men to go to.”

Bell and Dourish (2007) analyse the social imagery of sheds and shedding, identifying many aspects which connect them to hackerspaces. “Sheds are physically separate from homes, but make no sense without a home nearby; they are gendered male; and they seem to operate on altered schedules from that of the home.” (3) Sheds occupy a liminal position vis-a-vis the home activities of male private life which cannot be accommodated into the closed sphere of the family can take place. This is the perspective that unites the otherwise disparate range of activities which can take place in a shed, such as the storage and usage of hazardous chemicals and garden tools as much as pornographic materials, drinking and swearing, letting of the steam or prolonged concentrated handiwork, making them “a place of secrets and things best left unspoken” (3) While sheds are normally located far away from the work place, there is a complementary tradition historically cultivated by both manual workers and intellectuals to work out of sheds in the backyards or gardens of their homes.

(Wilson 2005) The ambiguous position of the shed as a multi-functional space which can accommodate missing elements of home as well as the world of work is in close alignment with the way hackerspaces have been argued earlier on to be situated outside of the modern institutional grid, being able to serve as sites for co-working, socialisation and hobbies too. I used the theory of third places to capture how such ambiguity complements the compartmentalisation of everyday life and creative energies that characterises modernity. Moreover, echoing what I showed to be the fear and fascination with hacking and hackers, sheds are often described as chaotic places resistant to notions of order and accountability that lack the usual segregation of activities. (Bell and Dourish 2007, 8–9) Lack of discipline, unauthorised access, repurposing for unintended usage are of course all well established notions associated with the amateurism and autodidactism of the hackers.

Wilson and Cordier (2013) provide a survey of Men's Sheds literature, giving an account of typical activities, which are woodwork, ironwork, and other forms of "light engineering" such as building bikes. Tinkering and bricolage clearly fit the profile of members who are most often former tradesmen and manual workers – now usually grandfathers – over 65 years of age. Other important target groups include (younger) unemployed, rural, or indigenous men. Studies such as Golding (2011) emphasise that the specific health and educational needs of these groups are coupled with a particular aversion to the institutional contexts of health and educational providers, going as far as to suggest that "learning scares away men". Similarly, Wilson and Cordier (2013) finds that literature on Men's Sheds promotes a "health by stealth" approach. (459)

Even though the first Men's Shed was founded in Albury, NSW by a Rotary Club chapter in 1978, (Garry, Matt, and Vicki 2008) the bulk of them were founded after 2005. By now a wide range of community organisations operate Men's Sheds and maintaining a shared machine shop is common to their setup. Answering to the growing popularity of Men's Sheds in Australia as well as the research findings cited above, the government has recognised their value and incorporated them into its policies. (Wilson and Cordier 2013) Complementing other funding opportunities aimed at these community organisations, in 2010 the Department of Health has allocated over three million Australian dollars to one of their national networks (AMSA) "to secure the sustainability of Men's Sheds and to fund the distribution of health promotion resources through Men's Sheds" (Health and Ageing 2010)

All in all, Men's Sheds introduce an interesting perspective for the appreciation of the social dynamics of hackerspaces and the historically ingrained practices that shape participation. They also serve as a model for involving social groups which are less privileged than the typical hackerspace participants, working with "low tech" and integrating in local community life. Above all, however, they go beyond the hackerspace model because they are organised by one social group to enroll another group, directing creative energies and work to gain benefits in the areas of health and well-being. This approach have been termed "health by stealth".

In contrast, hackerspaces are self-organised and self-managed spaces where creativity is undirected and labour is potentialy unalienated. The comparison with Men's Sheds brings out the differences between the civic impulse behind non-governmental organisations, supporting the hypothesis that hackerspaces cannot be understood adequately on the basis of the ethos of civil society.

8.7 Public Access Venues

Public Access Venues – libraries, telecenters and cybercafés – fulfill a similar function than hackerspaces in the sense that members of the general public can access ICTs and meet each other away from home and the workplace. A Telecenter is "a nonprofit venue open to the public, which offers ICT as part of its services intended to help community development." (Clark and Gomez 2012, 2) They exist in almost all countries, originating from smaller movements in Denmark (Electronic Village Halls) and the US (Community Technology Centers). Despite their Occidental origins, telecentres are most significant in poor countries where they fit into the development agenda professed by rich countries and leverage the funding coming from the ICT4D (Information and Communication Technologies for Development) industry: "ITU, IDRC, USAID are major international leaders in the telecenter movement"[102] (Colle 1999, 433) Telecenters are important providers of ICTs in rural and other underserved locations, while cybercafés are far more numerous and generally widespread in urban areas. It seems that even though libraries routinely provide similar services, users do not consider them serious contenders to the other two categories of public access venues for a number of reasons explained below. In the case of all three types of public access venues "users are generally young, educated and have moderate income levels." (Gomez and Gould 2012, 40)

It is interesting for a moment to think about hackerspaces in the context of venues that provide access ICTs for the general public, even if the latters' mission largely boils down to something as prosaic as an Internet enabled desktop computer rather than a well equipped shared machine shop. (Clark and Gomez 2012) Apparently cybercafés, libraries and telecenters have not even embraced FLOSS as an infrastructure option and making use of OSHW is far from their priorities: they mostly conceptualise their clients as Internet access consumers, not even as content creators or technology tinkerers. Even Internet access is often limited because libraries and telecenters routinely block porn and often block social networks even if "all people want is Facebook and porn." (Cited in Gomez and Gould 2012, 40) The authors argue that if public access venues want to stay

[102] ITU is the International Telecommunications Union, the technology regulator of the United Nations. IDRC is the International Development Research Centre, a state-sponsored Canadian development agency. USAID is the United States Agency for International Development, a state-sponsored US development agency. Similarly, "Vigorous actors in championing and supporting these enterprises are United Nations agencies such as WHO, ITU, FAO, and UNESCO, bi-lateral donors such as USAID and IDRC, and national governments from Hungary and Malaysia to South Africa and Canada." (Colle 2004, 5)

relevant and focus on realising their core mission – e.g. serving the needs of the local communities – then they have to reevaluate their relationship to what they see as "non-instrumental" uses of the Internet:

> More research is needed on non-instrumental uses of ICT, and porn in particular, and their relation to public access computing and community development, in order to better understand the challenges faced by public access venues that want to offer ICT as a contribution to social and economic development of underserved communities (Ibid.).

It is particularly ironic that despite libraries and telecentres enjoy state subsidies from home (libraries) and abroad (telecentres) to engage users with ICTs, in practice they underperform compares to their commercial counterparts. Gomez-Gould2012a claim that perceived "coolness" of the venue and "trust" in the operators are major factors that draw people to use ICTs at these locations. While libraries and telecentres are considered respectable and authentic places, they fail to provide a confortable and supportive socio-technical environment. The results of the global comparative research project focusing on typical users of public access venues summerised in Gomez (2012) show that the three most recurring advantage of cybercafés is (1.) their extensive opening hours; (2.) availability of food and drinks; (3.) unrestricted Internet access.

At this level of detail it is already possible to see some of the same problematics and characteristics which set hackerspaces apart from other genres of shared machine shops. Twentyfour hour access, self-service fridge and kitchen, as well as plenty of free Internet are three triangulation points which describe the particular sociality produced in hackerspaces quite precisely. To understand the coolness factor, it is enough to consider that libraries and telecentres, as much as Fab Labs and accelerators, will have vending machines with food and drinks and for instance designated smoking places. However, neither food provision nor smoking places will be arranged by persons who actually love to cook or smoke. In the same vein, we have seen that most genres of shares machine shops organise the social architecture of the organisation in order to direct creativity, exploit collaborative labour and leverage peer production practices for some end. The "non-instrumental" practices identified by Gomez and Gould (2012) which actually make ICT domestication work are nurtured in the hackerspaces while discourages in more institutionalised, commercial and recuperated spaces in order to encourage a specific kind of producticity.

The international comparison of public access venues show that while libraries and telecentres are organisations built from the ground up to answer to the needs of their users, the national and supranational institutional (development) agendas that drive them effectively prevent them to do so. On the other hand, there is increasing demand for and research to make telecentres (financially) sustainable, that would enable them to develop more autonomy and for funders to focus on other areas. Michael Gurstein (2011) who is the seminal scholar on community informatics argues that “this sustainability was a more or less complete pipedream which any realistic assessment of the circumstances of Telecentres would have determined” since telecentres were established exactly in the areas where market-driven cybercafés were unviable. (Zahra 2012) Meanwhile, cybercafés as market actors cater for the perceived needs of their users but only in places where the population can sustain them. As explored at length above, hackerspaces have the advantage that they occupy a protected niche between market pressure and institutional agendas, allowing for a freewheeling exploraton of agency between humans and machines – or ICT domestication, if you will. Compared to public access venues, hackerspaces are the wet dream of development agencies[103] – except that they mostly exist in well-served areas and cater for a highly privileged part of the population.

8.8 Conclusions

The round-up of other genres of shared machine shops aimed to situate hackerspaces in their family of organisations. It is clear that there has been a huge growth in similar organisations since 2005, something which is not necessarily appreciated by participants in the particular genres. I argued that the general tendency of shared machine shops is to direct the forms of creativity, sociality and technological potential showcased by hackerspaces into various channels – which necessitated moving away from the club model of self-organisation and self-management: essentially, *away from the peer-production of the organisation itself.* The most important observation from this panoramic point of view is that a wide range of social groups are enrolled in shared machine shops. The workers of Fortune 500 companies (in corporate hackerspaces), young entrepreneurs (in accelerators), university students (in Fab Labs), ordinary consumers (in TechShops), retired men (in Men’s Sheds) and the poor (in telecentres) are all mobilised to participate in shared machine shops while hackerspaces and makerspaces in some sense compete to provide a more laid-back environment for geek types. Many of these genres can be thus seen as recuperated versions of hackerspaces, implementing hybrid models which seek to exploit the economic (accelarators) and social (Men’s Sheds) benefits of peer production, but can only direct social processes to their own ends through implementing hybrid models of peer production, reinstituting various aspects of institutionalisation, alienation and hierarchies in their organisations.

These results underline the initial claim that hackerspaces are a privileged site for the study of peer production — and especially the peer production of hardware – because their sole aim is to facilitate peer production practices through an infrastructure that is itself produced and managed collaboratively.

[103] Hackerspaces provide all the colours of what Clement and Shade (1999, 36) call the Access Rainbow: carriage facilities, devices, software tools, content/services, service access provision, literacy social facilitation, governance

9 Open hardware case study I: The r0ket badge

In the larger scheme of things this first case study introduces the idea and the practice of open hardware as a form of peer production in the context of the hackerspaces. Based on these initial observations in the first case study, the second case study questions the concept of open hardware developed here and shifts the discussion to unfinished artefacts, and ultimately the unfinished architectures. Here I concentrate on three questions related roughly to my three main research questions. Firstly, building on the previous chapters on hackerspaces, I show how the values and diversity of the hacker scene (practically the only relevant social group involved in its production) is inscribed into the architecture of an open hardware device, differentiating it in shape, outlook and functionality from comparable consumer electronics such as a mobile phone. Secondly, I highlight the particular difficulties that open hardware production offers to grassroots research and development communities, and how the challenges transform the peer production model – open collaboration towards a common goal supported by ICTs. Thirdly, I scrutinise the advances and limitations in the democratisation of technological expertise that the r0ket project involved. At the same time I try to show how hackers appeal to, rearticulate, enroll and intervene in the more powerful social forces at play during what are often miniature technical decisions. In the course of doing all that I will need to explain what is the r0ket and how it works, along with its particular trajectory of production, distribution, usage and obsolescence. Therefore the order of sections roughly follows the life of the object from inception to obsolescence.

9.1 Tradition of hacker badges

If the r0ket was to be an archaeological finding uncovered near München, Germany in the year 2222 following the collapse of modern civilisation, future archaeologists may discern it as a small scale technological artefact from the silicium age. It would be treated as a very rare specimen of material culture and placed in the series of hacker badges found mostly in the territories of the descendant and decadent 21st century Occident. They would note that hacker badges first appear in the cultural and economical centres of North America, but some of the most sophisticated versions are found from years later in Europe. Some would perhaps suggest that these objects had little practical application since they do not appear to be compatible with the technological regimes and infrastructures prevalent at the time. A theory could appear that they served ceremonial purposes during meetings of an otherwise geographically dispersed engineering cult not very different from the masons of early modernity. Hackers, obsessed with codes, numerology and bad spelling, believed that they can magickally gain control of technology – and therefore society – by rearranging the components used in mass produced consumer goods into arcane patterns. Taken as material proofs of such analogical thought, the artefacts could be a starting point between the archaeologists of the future about the precise interpretation of technological determinism hacker badges embody: if the idea was that technology determines society, how they could have thought that some enlightened engineers can determine technology and by analogy, society? We have to address similar questions.

Indeed, badges similar to the r0ket appeared at hacker conferences DEFCON (2006) and HOPE (2008) in the United States, becoming a permanent feature of these gatherings. The story of the first hardware badge has already been bound up with the history of hackerspaces in particular, not only the hacker scene in general. In 2005 Joe Grand – also known as Kingpin – gave a Hardware Hacking Training at a Black Hat Briefings security conference in Caesars Palace, Las Vegas, NV. Kingpin has been a member of the hacker group L0pht Heavy Industries, whose hangout The L0pht (1992-2000) has been widely identified as the Ur-hackerspace in the USA [Farr (2009); BreAstera2008a 78-81]. L0pht Heavy Industries became known for their invited testimony to the United States Senate about the state of ICT security in 1998 where they claimed to be able to shut down the Internet in half an hour (Greenwalt and Pratt 1998). The Kingpin has been L0pht's resident hardware hacker and as many other members, he continued to be active in the scene. That year he showed off a custom PCB (Printed Circuit Board) he designed. Another legendary hacker, The Dark Tangent (Jeff Moss)[104] who is the founder of both the Black Hat security conference and the DEF CON hacker convention attended the training and asked him to design hardware badges for next year's DEF CON. The idea reflected a desire that the conference badge prepared for participants to show their names pinned to their chests should be interesting, playful and useful in other ways too. Reflecting the traditional hands-on attitude of hackers, their love of DIY technology in general, and celebrating the new enthusiasm over hardware hacking in particular.

As a modest beginning, the badge for DEF CON 14 (Riviera Hotel & Casino, Las Vegas, NV from August 4th to 6th, 2006) featured two LEDs that could blink in different patterns. Blinking LEDSs can be considered the Hello World application of hardware hacking. But these features were only there for participants to plan with: as a taste of things to come, in the first conference Shagghie (Scott Scheferman) fed the light of the LEDs into a custom-built analogue synthesizer to generate sounds based on the pseudo-random pattern of the two badge LEDs. Kingpin declared that the motivation behind making the first hardware badge was to promote hardware hacking which was "not well represented in the hacker world" at the time. The hardware badge quickly integrated into the intricate DEF CON tradition of puzzles and competitions. Every year a kind of treasure hunt is organised at DEF CON amidst layers of security systems with strategically placed vulnerabilities which over the years outlined a veritable encyclopedia of mathematics, security and signals intelligence history, from Egyptian hieroglyphs through musical notations to rotation ciphers. These games provided ways for

[104] Interestingly both hackers named here are 40 years old at the time of writing, working as security consultants.

participants to distinguish themselves from the crowd. Kleif and Faulkner (2003) argue that the particular pleasures (male) engineers experience through their work is tightly bound up with experiences of an encounter with the unknown, mysteries and feelings of insecurity – but an insecurity that can be at least theoretically overcome through struggle with technology. Finally making things work and mastering their mechanisms provides a feeling of being in control: something that the authors claim is not a common feeling for their subjects in other spheres of their lives. Indeed, participants in DEF CON puzzles describe their experiences in terms of frustration, sleep deprivation and craving (Elegin 2013). For example the winning team members claim to crave for devouring human flesh when the last step of the DEF CON 21 involves presenting the results on red paper to the judges:

> Once we calmed down, Beaker took off to get @”%#@ red paper. I am not sure what happened all I know is that I heard screaming, bones crushing, and what I can only guess was human flesh being devoured (I try not to think about how he got the paper). By the time he got back with what looked like freshly soaked red parchment (pretty sure he made it from the flesh of someone), we already had the answer (MLF 2012).

It is quite clear that we are dealing with a particular performance of masculinity. Interestingly, these games highlighted and developed a third use of the badges as a badge of honour, similar to military badges or digital badges. Military badges are awarded to army personnel for courageous conduct and other memorable achievements, while digital badges are awarded for the contribution to a peer production project for instance by Mozilla Foundation or Wikipedia users. At DEF CON there were two competitions: one to solve the puzzle and one for the most imaginative user modification of that year’s conference badge. As far as I could gather, the winners of both competitions are awarded with an Uber badge which lets them attend all future conferences for free, which incorporates a fourth use case for hardware badges in the hacker scene. While DEF CON probably evolved the most sophisticated hardware badge culture in the hacker scene, all four usage patterns encountered here can be found to more or less extent in other hardware badges like the r0ket.

To summarise, the analysis yielded four ways in which hardware badges at hacker conventions are culturally significant and some sense useful. First, as a traditional conference badge: to display the name of the participant on their chest along with optional metadata like affiliation, contact address, interests, etc. I would call this function *identification* because it simply pins a name to a person along with a couple of attributes. Arguably this is still the primary use of hardware badges since most owners are not very much entangled in the other aspects, and even people who engage with the further dimensions of the device tend to also use it as a conventional conference badge. Second, as small scale electronic artefact especially designed for user modifications, e.g. hardware hacking. I would call this function *play* simply because it allows hackers to enjoy themselves and exercise their skills while entertaining others – both of which involves peer learning and teaching technological expertise. This is obviously the most relevant aspect of the cultural practices surrounding hardware badges, so that it comprises the focus in the exploration of the r0ket device. Third, as a medium of mysteries to unravel and possibilities to discover, the badge allows hackers to distinguish themselves from their peers. I would call this function *reputation*. While building a reputation is important for the *ethical economy* (Arvidsson and Peitersen 2013) of hackerspaces too, it is in fact not as consequential as in the political economy of the information security scene (including the Black Hat Briefings) where it is an essential asset for independent researchers competing for consulting jobs. What is more relevant for the identity construction of hackerspace participants is that the badge, being usually specific for each conference, simply shows that the person has been there – and often it also shows in which capacity, as described in the next aspect. Fourth, the badge is often the most discernible manifestation of an actual ticket (purchased or otherwise) which grants access for participants to an event in a certain capacity: for instance as organisers, attendants, volunteers or press. Given the logic of hacker culture which favours performativity, this aspect is not strictly utilitarian but enhances the cultural value of the artefact. On the one hand, it comprises an access control system akin to those we will encounter in the second case study. HighWiz et al. describes the Black Badges that can be won for hacking the badges or solving the puzzles at DEFCON:

> Black Badges are the ultimate award you can receive at DefCon. They allow you free entrance to DefCon for life. For a lot of people that's reason enough to dedicate yourself to getting one. For others it's a badge of honor and pride (2011).

On a final note, the micro-stories outlined above should be read against the broader background of the rising availability and popularity of hardware hacking, especially what I identified as *physical computing* in the years after 2005. I tried to show in chapters 6 and 7 how the rising popularity of hardware hacking is closely entangled with the establishment of hackerspaces: two tendencies which mutually strengthened each other. On a shallow phenomenological basis, hacker badges have been arguably the most visible signs of these tendencies observable in terms of material culture at hacker conventions. As I argued in Chapter 4, the endemic nature, origins and use of hardware badges makes them a privileged site for the investigation of open hardware production in the specific context of hackerspaces.

9.2 Peer production and distribution

Two batches of r0kets have been made: a first run of 3000 for the Chaos Communication Camp (2011 August) and another 1000 for the Chaos Communication

Congress of the same year (2011 December). Here I relate the making of the first batch in more detail, and then follow up with a summary of the differences in the making of the second batch, closing with drawing together the analytical conclusions accumulated throughout the section. I focus on the problems the project offered to participants and how the collaborative environment and productive infrastructure of the hackerspaces helped them to face these challenges.

Following the methodological principle of historical and geographical specificity, the story of the r0ket badge can also be read as the story of the appropriation of the badge idea that started in the USA by European hackers on the other side of the Atlantic. I have showed earlier that the idea of hackerspaces travelled in the opposite direction: the US Hackers on a Plane picked it up in Europe and popularised it in North America. Prior to that, the ideas and practices of hacklabs have developed into the ideas and practices of hackerspaces as their centre of gravity moved from South Europe towards the North. Once the r0ket story unfolded, I will return to this issue to substantiate how the appropriation of the hardware badge changed the ideas and practices involved.

The r0ket emerged from one of the most active local chapters of the Chaos Communication Club: the München chapter whose base is the µC3 hackerspace. Creators of the r0ket device first encountered the idea and practice of the hackable hardware badges at DEF CON. They wondered why European conferences lacked these contraptions which add excitement to hacker conventions. They thought that it is a good way to introduce the vast numbers of attendant programmers to the world of electronics. However, they wanted to go beyond the one-off, show-off qualities of USA hacker badges and create a tool that can be used in hackerspaces as a general purpose programmable microcontroller development platform and prototyping board (mh et al. 2011). It is important to note the changing landscape of hardware hacking: when the Kingpin started to make hardware badges for DEF CON, practically the only known hackerspace was his old groups' old hideout — but since 2006 hundreds of hackerspaces have been established both in North America and in Europe. Moreover, most of the new generation hackerspaces had an overwhelming majority of software developer members, but a collective enthusiasm for dabbling into hardware design. While the first DEF CON badge was ahead of its time, the r0ket could count on an established audience hungry for learning more about electronics or work in the environment of embedded microcontrollers where they could write low-level code that is closer to the metal.

The concrete apropos for taking up the project was that the annual "Workshop Weekend" called Easterhegg – a sort of little sister of the Chaos Congress organised each year by the Club Computer Club on a volunteer basis in a different city during the Easter holidays since 2001 – was coming to their city, München, in 2010. Hackers at µC3 wanted to contribute to the event. A few members of µC3 have put together a hardware badge for the small Easterhegg event which was distributed in 300 copies and the 100 remaining "sold out within minutes", so they were really happy with the response from the community.

They later reflected that the success of the badge also enabled them to learn about the problem and get more confident with their abilities (sec, r0ket hacker, personal communication, 2011 December).

9.2.1 Making the r0ket

Empowered by their first venture, they embarked on a second quest for making a hardware badge for the Chaos Computer Club's flagship event, the Chaos Communication Camp. Here it is interesting to highlight that the Camp is the pinnacle of fun in the hackerspaces scene, where attendants make the most effort for contributing to an interesting and inspiring event. While the annual Chaos Communication Congress enjoys more prestige, it is a more official event where hackers present new vulnerabilities, research results and discuss techno-politics. The Club makes a profit from the Congress which they can use to subsidise the substantial costs of the Camp. Since the Camp is organised every four years, hackers feel more special about it.

The work began around four months before the event and progressed with exponentially increasing speed until the cathartic moment of release. I talked to team members and watched them work on the artefact, but the git repository[105] hosted on github.com (r0ket contributors 2011) gives a blow-by-blow account of the development process. Two young individuals studying relevant IT topics at the local university and sometimes working commercially as system administrators lead the hardware and software development (Schneider and Stefan "Sec" Zehl, respectively). Their work was complemented by a circle of about a dozen individuals contributing in diverse ways from coding to cleaning. At later stages when more workforce was needed, even more people joined the effort, so that it is reasonable to assume that as much as a hundred people participated in the development process before the launch of the r0ket. As it will become evident in the next pages, starting from the hackerspace in München, the r0ket project created an extensive network of enrolled participants, collaborators and business partners ranging from individuals through nonprofit foundations and commercial companies. Of course, as the r0ket have been produced and reproduced, distributed and the user base grew, the network grew exponentially – only to collapse in a few years as the r0ket went out of fashion.

As Sec and Schneider, most of the core contributors were university students in München, some of them already working in the commercial sector. Despite their other obligations most could find ample time to be active in the Chaos Communication Club and its local chapter hosted at µC3, including supporting the r0ket project. Gender ratio lingered around one to ten, so that some of the core contributors were women, which is consistent with reports of the Chaos Communication Club in general (Blanc and Noor 2011) and the Chaos Communication Camp in particular (Braybrooke 2011). At the same time these numbers should

[105] The concept of distributed revision control systems like git is introduced in the earlier in Chapter 5.

be taken as a snapshot at a particular time (2011) since informal discussions consistently agree that the ratio of women who participate in the hacker scene is consistently growing year by year. I would point out three factors that may play a role in the growing number of women and transgender participation in hacking. Once, ongoing historical process of the mainstreamisation of hacker culture leads to more variety and so more inclusion. Twice, the verso of the same process is the resistance to mainstreamisation which leads to a dynamics of internal critique, bringing in a strong stream of awareness to various privileges articulated in the hacker scene. Three times, possibly as an outcome of the two previously mentioned tendencies, the resurgence of OSHW hacking and its inclusion in the core of the hackers' technical repertoire seems to bring in more women. In fact many of the high profile hackers in OSHW are women inventors, teachers and entrepeneurs running successful SMEs, such as Limor 'Ladyada' Fried who runs Adafruit Industries, a design practice, webshop and knowledge base catering for hobbyists and hackers.[106] Still it has been observed that there are even less women in FLOSS development than in the strictly commercial sector (Nafus 2012). However, the results of the latter study are probably skewed by the fact that while the hacker scene proper is comprised almost exclusively of engineers, industry positions cover a much wider scope from copywriting to quality management.

All in all, these results confirm that hacker culture is still strongly marked as a masculine, even if hacker masculinity is very different from mainstream machismo. The phenomena of Men's Sheds described in section 8.6 provides an analytically compelling synthesis between mainstream forms of masculinity and the tinkering engineers' subcultural form of masculinity. The repeated rise of women in the hacker scene (since they were initially the majority, before programming and electronics became privileged positions in the job market) and the social conflicts that result are documented and analysed in Toupin (2014) who writes about the wave of gender-oriented hackerspaces which were established in the last few years following controversies and repression of women in mainstream hackerspaces.

It should be appreciated – especially if we are interested in how peer production of open hardware works in the hackerspaces – how the initiative of one or two people becomes a collective project. The r0ket was conceived by few, but taken up as a collaborative effort by many members of the hackerspace, and then embraced by the larger hacker scene in which they were embedded. One striking aspect is that what makes an idea like the r0ket attractive for contributors is precisely that it is a project that requires a lot of contributors. Hackers are attracted – amongst other things – by the sheer technical and organisational challenge. Therefore, a successful project have to be sufficiently difficult and ambitious: a rather paradoxical notion.

The work process developed from initial discussions through semi-regular physical meetings in the hackerspace to a contributors-only mailing list. As the

[106] The well positioned Open Source Hardware Association is an almost all-women organisation, see the profiles on http://www.oshwa.org/about/our-team/

hackerspace began to fill with components, more and more people got excited and involved in the project. This was very welcome since it quickly turned out that beyond writing the firmware code and designing the hardware layout, other equally large parts of the project had to be covered. Once, a *business strategy* of sorts had to be managed to round up investors, controlling the cash flow all the way from seed-funding the idea through buying the parts to selling the product. Twice, the bill of parts had to be *sourced*, which means looking up and negotiating with suppliers. Thrice, *mobilisation* had to be done in order to convince supporters to enroll into the network of the project and raise the expectations of potential audiences. In terms of work hours these have been just as significant in the energy spent on the project as the more technical tasks of software and hardware development. Moreover, the hardware design and therefore the functionality offered by the software had to be constantly redesigned and rethought as new possibilities opened and already established conditions dissolved. Larger firms often dedicate separate departments – such as sales, acquisitions and marketing – to these activities.

However, in the case of the r0ket device, a concrete business plan never materialised — it have been figured out in practice through endless meetings of continuously changing ideas, successes and failures. New possibilities allowed designers to add more parts and functionality to the device, while failures forced them to return to a more limited plan. As the process rolled on, certain choices cemented and the space of possibilities shrunk.

In any case, retroactively it is possible to identify the three most important elements of the business strategy that finally made for a viable project. These possibilities were known quite early in the process and factored in the subsequent decisions about how to navigate between financial hazards and technical challenges. The first was the ability of µC3 to enroll the Chaos Computer Club in the project. Taken the European hacker scene as a whole, the CCC was probably the only organisation that had enough cash to secure seed funding for the r0ket. The Chaos Computer Club had both reputation to lend and cash to offer up front to the r0ket hackers. While social capital can be turned into financial capital – and it has been, as described in the next point, it has to be emphasised that in this case both were strictly necessary. There was no time or energy for running a crowd-sourcing campaign, for instance. Cash was needed for sourcing the parts and it was needed quickly. On the one hand, µC3 as the local chapter of the Club could officially turn for the organisation for help. On the other hand, several members of µC3 have been deeply involved in the organisation of CCC events and processes. Furthermore, the µC3 already had a history of highly visible electronics projects that contributed to the prestige of the organisation and the fun of hackers (for instance ACAB, see below). I describe these ties in more detail when discussing the distribution phase later on. What is important here is that enrolling the CCC in the project was not difficult because µC3 and its crew was in many ways part of the Club since a long time. Therefore, the r0ket became an official CCC project[107] as the official badge for the 2011 Chaos Communication Camp. Each board is signed on the backside "designed by ccc". The financial agreement was that the Club pays for most of the manufacturing cost of the devices, and builds their cost into the ticket price of the event.

The backing of the CCC gave an air reality to the project, but it was not enough for making a device that is more than a toy – a device that can be useful later on as a development board. The project needed more sponsorship. Luckily, a major microchip vendor was ready to flood the market with a new type of CPU that needed positive marketing. NXP Semiconductors was ready with their ARM Cortex CPUs that were much more powerful than previous devices, and the first in their category to go beyond the "embedded devices" market segment. The new chips could be put into mobile phones, small computers, etc. So they were happy to give the chips for free for the hackers to play with: basically the cost could be covered from the company's marketing budget. The CPU was the most complex and hence most expensive part of the design, so the r0ket got a new and interesting component without having to do as much as putting the logo of the company in the documentation. NXP knew that hackers will look up what kind of CPU is in the device anyway. In a way the r0ket ushered in the ARM Cortex era – an architecture that can be found by now in most households of the developed world (built into mobile devices).

On another note, the case studies here are situated in a period where the basic microchips used in hackerspaces shift from Atmel's AVRs that power the Arduino microcontroller to NXP's ARMs which power the Raspberry Pi. The next case study outlining the development of door systems will often feature older systems built with Arduinos and newer ones sporting Raspberry Pis, as well as the mix of the two. If the Arduino with its Atmel AVRs enabled or at least determined the emergence of hackerspaces, the Raspberry Pi with its NXP ARMs ensured their mainstream success. From the point of view of competition between capitalist firms, the r0ket was instrumental in propelling a new product and a vendor into popularity within this small but strategical market segment that is the hacker scene. The desire of µC3 hackers to make an amazing device for popularising hardware hacking fitted in with the plans of a major electronics corporation to capture a part of the market. The move illustrates the pragmatic aspect of hacker culture, when they make do with what is possible in the present situation in order to get results, instead of striving for purity even if that means that actions remain purely symbolic, as some activist groups do.

The last element of the r0ket business strategy was to find a place to do the PCBs (Printed Circuit Boards). A circuit board is the usually pale green or bland brown surface where electronic components are mounted. The components are connected by glittering streams of metal to each other on the board. Schematics are the plans which specify how to connect which components to achieve the

[107] The project is available under a subdomain of the Chaos Computer Club (http://r0ket.badge.events.ccc.de) which now redirects to the domain acquired later: http://r0ket.de/.

desired functionality. The manufacturing phase actually involves two main steps which have been performed by different companies. One is the PCB manufacturing that results in the boards themselves, and the other is the "pick and place" process where the board is populated with the actual components. The Bill of Parts is a list of the components that have to be sourced, picked and placed on the board. For PCB making there are many methods for DIY PCB making using acid, ultraviolet light and so on – but none are well suited for mass production on the scale of a thousand devices. Therefore even hackerspace members often send their schematics to factories which accept small-scale orders. Outsourcing the PCB manufacturing to a company also makes for more precision which can be important since one broken connection can completely disable a device and it is often hard to determine what is at fault. Hackers could spend nights patiently going through each connections between each component with a multimeter, measuring if the path between them is conductive or not.

The r0ket team decided to use an SME called LeitOn which is headquartered in Berlin. They simply choose the company because it offered the best price. The only problem that occured with the manufacturing process was that the original specification highlighted parts of the green PCB in a ripe yellow colour, a design decision which was not implemented by the comany. LeitOn later sent a couple of extra r0kets to the team just to prove that they are technically able to respect these requirements.

Fortunately for the r0ket hackers, they had friends at an SME which had a pick and place line and agreed to sponsor the project. A deal was struck with E.E.P.D. GMBH that the company would be stopped for a day, paid work put aside, and the r0ket PCBs would be populated using the full capacity of the company for a nominal price. First generation r0kets therefore bore the "assembled by eepd.de" legend on the back.

As r0ket hackers emphasised, these three elements – (1) seed funding and backing from the Club, (2) donated CPUs from NXP Semiconductors, (3) free pick and place service from E.E.P.D. – along with hundreds of smaller donations and contributions, were essential for the realisation of the project. Therefore, we can safely state that making the r0ket at 30 EURs per unit was a commercially inviable project, and this was clear to all parties involved in the process. The price of physical manufacturing and sourcing is a particular difficulty during the production of OSHW which sets it apart from FLOSS. The r0ket device shows how collaborative networks, reputational dynamics and community values could be leveraged to enroll individuals, foundations and even corporations into the project. Concentric circles of reputation have been at work in this case: the wider hacker scene as a potential market of first adopters or promoters of a product in relation to capital, the Chaos Communication Club as a respected hacker organisation in relation to the particular vendor, the µC3 as a capable hackerspace and an active local chapter in relation to the Club itself, and the reputation of particular members whose work contributed to building up all the other reputations so in turn they could finally leverage them.

Moreover, it is worth noting that the particular role of the Club in the production of the r0ket was following the pattern described in Chapter 6 in connection with the insitutionalisation of the hacker scene. As a formal entity that promotes the values of hackers and represents their interests, the Club could be used to negotiate between larger social forces (such as corporate manufacturers) and the hackers (such as the r0ket team). In this capacity it acted as the medium through which the relative autonomy of hackerspaces could be exercised: to engage with market forces but on the terms set by the hackers.

When it comes to the demise of hacklabs, it is easy to see that the autonomous movement which provided their political context would not have tolerated any negotiation with capital, especially if it is not about the survival of the community but about building some toy-things to play with. Therefore, the hacklabs model did not allow the addition of OSHW production to the already existing FLOSS production and hardware recycling repertoire of Shared Machine Shops. The reason was simple: OSHW production is more capital intensive and the political economy of hacklabs did not allow for mobilising external sources or to acquire enough members from the middle class to subsidise the expansion of the field of activities.

Another point of comparison between OSHW and FLOSS in the r0ket case is the critique of intellectual property. Relative autonomy seems to be able to accommodate paradoxes such as the above where hackers are happy to accept CPU donations without asking for the blueprints of the CPU, effectively upholding the intellectual rights for the company to keep the working of the CPU secret, therefore holding on to its monopoly as its sole manufacturer. As observed before, OSHW is seldom completely open source: even at its best it is merely composed of widely available off-the-shelf parts. Of course, even if the hackers would have demanded the schematics and manufacturing recipe for ARM Cortex, effectively breaking the monopoly of the manufacturer on the product, they could not have been able to reproduce it themselves. Even though there is much to learn for independent researchers like hackerspace members from the schematics and manufacturing recipe of a microchip, they lack the means of production to actually make a microchip. Unfortunately the industrial scale of contemporary microchip factories is simply out of league for hackers, even if they are cooperating on a global scale. Therefore in practice the r0ket team was taking diametrically opposed stances on intellectual property. On the one hand, they were happy to help a company which made proprietary hardware, lending legitimacy to the firm through their OSHW project, which the firm reciprocated by donating the CPUs. On the other hand, they asked the PCB placement factory to support an OSHW project that is at odds with proprietary hardware, questioning the legitimacy of proprietary hardware patterns. Presumably, the PCB factory would not have lent the r0ket team its populating capacity if the team worked on proprietary hardware, because the whole idea was to create a device for everybody to modify which enhances the practical and theoretical capabilities of all hackers. The business model behind providing pick and place services does not depend on whether the plant is processing open source or proprietary designs, and presumably its operators like electronic artefacts – so it makes sense that they supported the initiative.

Even more striking is that during all these negotiations no discussions of licencing took place. Following the example of the Kingpin, the creators published a detailed documentation of the badge. Sharing the schematics and the bill of parts with the public have been enough to qualify as OSHW, which the r0ket team topped up with the assumption that they will be actively supporting hacking the device, sharing their knowledge with anybody who cares to ask. But no specific conditions for licencing have been made, which is very different from negotiations and controversies around FLOSS where debating licences is one of the sharpest points of contention both between hackers themselves and in their interactions with capital. For instance, the classic free software vs. open source debate revolves around the "sticky" part of the GPL (General Public Licence) that prohibits combining GPL licenced code with proprietary parts. Therefore, GPL is considered a restrictive open source licence, and a veritable free software licence. However, for the reasons explained above, it would be practically impossible to make "free hardware" according to the principles of the GPL, because most useful OSHW includes proprietary pieces in its Bill of Parts. But such nuances have not even been part of the discourse. It is hard to discern the exact licences of the r0ket components, even though it seems that the firmware is licenced under a very permissive BSD licence[108] and the licence for the hardware is only visible in the CAD files: CC-BY-3.0. The latter licence allows users to do anything they want with the content as long as they credit the original authors. Ironically, the original authors are not identified in the files themselves and ambiguously identified in other sources. Even more ironically, the design files can only be opened with proprietary software: the state of the art Eagle PCB design tool (layout editor, schematic editor and auto-router). When it comes to designing complicated schematics, Eagle have been used by even the most ardent FLOSS advocates since there is no comparably powerful free software alternative. The most notable contender which arose since the creation of the r0ket is the KiCad software suit, licenced under the GPLv2 and developed at CERN.

In summary, it is fair to say that at least in this particular OSHW case the licence is not as important as it is in FLOSS development, where it occupies centre stage in the identity construction of participants. One reason could be immensely practical: most open hardware – definitely including the r0ket device – is not industrially relevant, therefore its licences are unlikely to turn up in court any time soon. Another practical reason is that hardware is not as easy to document and reproduce as software, and therefore it is common to ask help from the original designers when reproducing or developing a piece. A third reason is somewhat contradictory to the two others: once the designers publish the essential information about the OSHW design (mainly the schematics, bill of

[108] Found under http://r0ket.de/software and https://raw.githubusercontent.com/r0ket/r0ket/master/firmware/LICENSE.

parts and firmware), it is hard to stop anybody from trying to reproduce their results, irrespective of licences. *In OSHW more things depend on the attitude and practices of the makers than on the legal regime they choose to bring into the project.*

The answer – why licences are not as central to hardware hacking as they are to software development – could also lie in the shifted trajectory of the latest wave of hardware hacking. As I argued in Chapters 6 and 7, the rise of hackerspaces which are founded on the idea of hardware hacking has been partly inspired by disillusionment in software development, or at least in the fact that it has become so common and thus mundane, regulated and ruled by major corporations that it did not present the excitement it could around the turn of the millennium. As a result, the low tide in software development is matched by a high tide in hardware development – and perhaps the centrality of legal discussions have been a particular property of the rise of FLOSS. Such hypothesis finds some support in the fact that licencing discussions so central to hacker discourse only a decade ago have died down now. Without theorising in turn the reasons for that, it is enough here to refer to the fact that the licencing landscape have stabilised and all the arguments have been played out in favour of the particular licencing choices – so even if two hackers disagree about their taste of licences, both can reliably reproduce the arguments of the other, crystallising debates to a ritualistic form that is technically and ironically called *religious wars* in hacker parlance.

In this respect the situation of hardware hacking is similar to the situation of software hacking before the social conflicts that lead to the emergence of FLOSS licencing erupted. The corollary of such an argument is that OSHW will reach maturity in a number of years and that will be the time when participants will (have to) take licencing more seriously. However, the example of the RepRap project mentioned in the previous chapters presented shows that it is hard to know where community-based projects will end up. While RepRap participants were enthusiastic about a small spin-off company emerging from a hackerspace, they began to question their previous stance when the company received venture capital backing filed for dubious licences, only to begin suing OSHW practitioners after being acquired by their greatest enemy, the multinational corporation Stratasys. In any case, the r0ket drew a very different and one could say more typical trajectory: no spinoff company was born (although most conditions were met – more on this later) and after an initial period of excitement is slowly faded into oblivion.

9.2.2 Sourcing

Sourcing the parts have been the second major challenge for the r0ket team. Sourcing is a major part of OSHW development, and an under-appreciated one. It involves research, negotiations and deciding on trade-offs between price and functionality. Most hardware project have to match a target price per unit – a price that grows constantly and have to be kept under control. The more units are produced, the more important is to keep a tap on the price of the parts, and the more concessions it is possible to get from suppliers. Knowing the right sources can decide on the life and death of the project. The r0ket team was in an especially difficult position in this regard since they never dealt with such a huge run of several thousand units so they had little experience. However, they did small projects before where they did face similar problems. They were especially fond of the OSHW principle of using off-the-shelf components. One element highlighted in several interviews was the Nokia 1110, the classic LCD display of the early Nokia mobile phone models. They argued that it is a part commonly found in hackerspaces. Since it is used in many older phones, there is a host of Chinese manufacturers making knock-offs that can be easily and cheaply ordered over the Internet. All these considerations come into play in FLOSS development too, but they never received analytical attention partly because they are not so important, partly because they are not so visible and explicit, and partly because practitioners themselves do not put a lot of emphasis on them. Therefore we can say that moving from software to hardware production sourcing simultaneously moves towards the centre of the stage.

FLOSS developers begin a project by looking around for existing technologies that do any part of the job. For instance they often shortlist, test and evaluate multiple software libraries that include the necessary subroutines. They are sad not to have found anything useful or amazed at the elegance that a library they discovered is addressing a particular problem. They make choices based on their taste in engineering and the size of its user base as well as the number of developers. While the first one (taste) is a highly subjective choice determined by which tools the hackers like to work with and what are the technologies that they are familiar with, the latter two (the size of the user base and the number of the developers) are used as a proxy indicator for a number of other factors which are beyond the control and often the grasp of the hackers. These factors are the maturity of the library, its stability and quality, but also the frequency of security updates, the responsiveness of the developers to queries from the users and the prospective lifetime of the library. For instance many a FLOSS project have faultered due to upstream developers abandoning the underlying libraries. A larger user base means that many people rely on the resource so their is a higher chance of somebody stepping in to carry on the work of the library's developers in case they decide to do something else with their lives. But most often programmers are amazed by the work other people have done before them and they feel that they are standing on the shoulder of giants. The simple fact that FLOSS hackers can reuse the best components in their field without any cost and any difficulties creates a feeling of community, even communism amongst them. They are effectively at the mercy of each other's quality of craftsmanship, hence the high standards imposed by the morality in their midst. While this is true for OSHW hackers to some extent too, during sourcing they are mostly amazed by the low prices and versatility of Chinese manufacturers, which makes them realise how well the free market works. Therefore in my experience of speaking to both FLOSS and OSHW hackers, the latter are somewhat less oriented towards working with each other than the former, simply because most of the components they use come directly from vendors than from other developers with a face.

I argued in Chapter 5 that even though high profile FLOSS projects are driven and largely developed by transnational corporations so much so that the history of FLOSS is effectively a history of those corporations, the bulk of FLOSS projects are actually developed by a small number of contributors or even individuals who are not (directly) employed to work on them. Moreover, the historical trajectory of FLOSS followed a different arc than that of OSHW. Even though companies were there from the beginning, the relative autonomy of hackers was articulated more successfully because peer produced technologies did not become an integral part of capital accumulation models since the beginning. What happened with OSHW was very different, since capital investment in relatively popular OSHW projects was both necessary and available from the beginning. Therefore, OSHW evolved in a tighter symbiosis with capital interests in the way that Hess (2005) argues in reference to what he calls *object conflicts*, when the community is convinced that their goals can only be realised through securing the cooperation of market actors.

In terms of the differences between the peer production of software and hardware, it is more important how these differing conditions restructure the production process. Hardware hackers evidently have to cultivate a much closer relationship to the commercial sector – and often the public sector too. Therefore there is a larger surface of contact that can lead to frictions, or alternatively enhance the grip of capital on the engineering subculture. Examples of the former are hacking the Microsoft XBOX and similar consoles documented by STS scholars, which created legal troubles for the hackers but also expanded the use cases and therefore the market of the relevant commercial products (Flowers 2008; O'Donnell 2014; Huang 2013). Examples of the latter are the recuperation of RepRap by MakerBot or the recuperation of DIY Drones by 3D Robotics which are discussed in section 7.1. In the two cases of the r0ket badge and the door systems it seems that hackers mobilise corporate resources through leveraging their relative autonomy so that their plans are less moulded to the requirements of capital. The fact that these OSHW projects do not produce a commodity for the market can be seen as their practical failure and ultimate insignificance, yet in the context of the hackerspaces they could be also considered *appropriate technologies* which fulfil the needs of their users (Schumacher 1973).

9.2.3 Mobilisation

Mobilisation was a crucial part since not all of the physical work have been done in the factory. The soldering sessions after the boards came out of the factory with most parts already fixed on the PCBs was related in the most excited terms by r0ket team members – since it made clear the large number of devices and the many people who would help out in the quest. They mentioned that some mistakes were done during manufacturing which had to be corrected by "a thousands of helping hands"[109] and that a few elements like the m0dulbus headers (explained later) were always meant to be soldered individually by volunteers. Calls for help have been sent out through the populous mailing list of the hackerspace and mobilisers crawled through chat channels where hackers hang out to lure them into µC3. About a hundred people gathered over a few weeks to participate in the 24/7 soldering sessions, some of them coming from other cities and many of them learning the skill of soldering on the spot. Even though the core r0ket developers were presently surprised and greatly humbled by the experience of such a successful mobilisation, they still did not have enough collaborators to match the amount of work to be done, so they ended up working long shifts themselves in the hackerspace. One participant proudly stated that "I probably held most r0ket devices in my hand during that time".

Another related anecdote was about how r0kets proliferated throughout the hackerspace, in a material approximation of how the project mobilised all the human and material resources of the organisation. Stacks of r0kets in various stages of completion filled all available space in the lab. Once a film crew came to record an interview in the hackerspace. Since the r0ket team at that point was not ready to reveal the details of the r0ket project to the public, they wanted the r0kets out of range of the cameras. However, they say it was hard to find a spot in the hackerspace without some r0kets coming into the picture.

Which were the conditions which enabled the OSHW hackers to rise to the challenge? In particular, what are the particular infrastructural conditions in the case of the r0ket device that enable the peer production of OSHW, the infrastructural factor that is so important for Kohtala and Bosqué (2014), Troxler (2010), Kostakis, Niaros, and Giotitsas (2014), Kera (2014), Seravalli (2012) or Lindtner (2014) (see 7.2)? I list five factors that have been highlighted by practitioners who worked on the r0ket, mixing subjective and objective factors as adviced by Seravalli (2012).

First, they already had a well equipped electronics lab with soldering irons ready, and a number of members who knew how to make use of those. While evident, this is an important point because it is the most fundamental raison d'être of hackerspaces: to provide a collective shed with more space and people than what members can afford to dedicate at home. Second, they had instructors and instructions ready to teach soldering and therefore involve more people in the operation. Here again the ubiquity of the "Soldering is Easy" fanzine (Jeff Keyzer 2011) which have been produced by some key participants in the hackerspaces scene is important to highlight as an in-bread schoolbook or instruction manual that integrates well to the engineering subculture of hackerspaces, because it assumes that anybody can learn soldering. The assumption that its readers are not specialists makes it an especially friendly introduction to soldering. Third, the r0ket team could leverage the existing communication tools already in place

[109] Probably a parabolic expression.

at µC3 for internal coordination and external outreach: mainly the IRC channel ("whole muccc is very much irc-based" wrote sva, a µC3 member, personal communication 2015-04-18) and the mailing list with hundreds of subscribers. Fourth, the community and reputation built around µC3 which enables mobilising collaborators to help in the soldering who were not directly connected to µC3 as members or otherwise. Here we can note that the hackerspace allows for a certain anonymous aggregation of reputation. In the hackerspaces scene particular contributions which are recognised by members of the community are not necessarily attached to the names of the contributors, since they may be many – rather, they are attached to the particular hackerspace. Therefore, members can count on the reputation of their space even if it has been (at least partially) built by other members. Fifth, the ample stock of Club Mate – a carbonated energy drink popular with hackers, derived from mate tea manufactured by Brauerei Loscher near Münchsteinach in Germany – allowed for non-stop soldering sessions by keeping hackers awake and concentrated.

What was exactly enabled by these factors in the context of peer production involved education, research and manufacturing. First, the potential of the ARM Cortex CPU have been explored through the creation of a material artefact that in true hacker fashion searches for the limits of that technological system, showcasing its possibilities. It is a venture of research in the sense that the particular CPU architecture have seldom been used in such an open-ended and interactive way before, and the product enables users to take the mission further. Interestingly, this was the interest of the NXP Semiconductor in the project. Second, the peer production process allowed the r0ket team to realise their primary objective: to draw more hackers into hardware/electronics, not well represented in the hacker world (in the words of Joe Grand). Even before distributing the r0ket, they taught a number of people how to solder, which is obviously the definitive initiation to hardware hacking. Third, while teaching and learning in a collaborative way, these people actually produced three thousand pieces of small scale electronic artefacts, fulfilling their promise to the Chaos Computer Club to provide the first electronic badge for the most high-profile European hacker conventions. While I highlighted the self-declared interests of the various stakeholders in the venture above, it is fair to assume that they met on common ground when they cooperated on the r0ket OSHW project: each of them were happy with the realisation of most of these auxiliary goals.

While it would be possible to frame these activities which are analytically distinct as peer learning, peer investigation and peer manufacturing or peer production, I argue that this would simply reintroduce the perspective of modernity which artificially separated these activities. Looking through the lens of everyday life in a hackerspace we can see how the peer production process spontaneously unifies these – once again, analytically distinct – experiences into peer production in general. In turn it is the general peer production process which involves components of peer learning, peer investigation and peer manufacturing as the sharing and therefore reproduction of old knowledge, the production of new knowledge, and material production in the form of tangible r0ket devices. *Peer* *production of OSHW involves the production and sharing of knowledge and material goods in a single movement unified by a particular sociality.* This theoretical statement explains how and why work and play can be seamlessly integrated in the context of hacker culture – i.e. because in a general sense they remain phenomenologically indistinct and culturally unmarked.

Therefore, it is fair to say that despite the division of labour between participants – from volunteers to core developers or from hardware to firmware hackers – the making of the r0ket device shows how peer production practices break down the larger division of labour in modern societies between education, research and manufacturing. I have attempted to argue in 7.3 that the hackerspace as an organisation escapes the compartmentalisation characteristic of the modern institutional grid established in modern societies. Extending that analysis, I here demonstrated it within the work process in more detail within the context of a case study.

Of course behind the tear-down of the division of labour established in modern capitalism we can also identify elements reminiscent of the transformations described by sociologists reminiscent of late capitalism – some of which actually point to the contradictions of the post-Fordist understanding of contemporary political economics. The most striking is the unfolding of the Terranova thesis of free labour as the basis of post-Fordist models of capital accumulation and value capture in our empirical results. In contrast with the insistence of Himanen (2001) that the hacker ethic is the negative imprint of the Protestant ethic as it has been proposed by Weber (1958) in his classic work, the opposition of capitalism and peer production is not very clear if we look at the details of their implementation. The r0ket team can be easily described in Weberian terms as following a vocation, a calling that establishes close ties between the subjective joy of hard work itself with a dedication to the good of the community and the reproduction of a higher moral order. The good hacker is known for her hard work, contribution to the community and commitment to the engineering ethics and aesthetics of the FLOSS/OSHW movements.

As we have seen, the incredible efforts which went to making the r0ket can be understood as unalienated labour which sidestep the division of labour and the antagonistic relations between the owners of the means of production and the workers who use it produce actual value. The goals and objectives of the project were eminently non-commercial: as it will become more clear later, the r0ket team exhibited a total lack of interest in the creation of exchange value, and concentrated on the creation of use value for a particular community with which they identify. However, in the greater scheme of things the production process played out in the context of larger circuits of capital accumulation. For instance, the Dutch NXP Semiconductor used it to promote their new CPU architecture offerings, in accordance with the post-Fordist thesis that in central economies product value comes from brand recognition rather than use value, while on the semi-periphery where actual manufacturing capacity have been outsourced, a host of Chinese manufacturers catered for the long tail of small-scale purchases through their online shops, just as Anderson (2006a) professed.

The critiques of the old capitalism would be happy to see that direct value extraction through alienation and exploitation features little in the production of the r0ket device, and if it did, then it did not take centre stage but occured on the peripheries of the network where the engineering subculture of hackers interfaced with larger social structures such as the globalised free market. However, critiques of the new capitalism would agree with the notion of Boutang (2011) explored in the theoretical introduction, who observes that novel ways of capital accumulation are based on externalities – in other words, they leech on processes that do not directly create exchange value. Therefore, *in order to mount a challenge to the latest capitalism, it is not enough to produce a commons. The question is how to proliferate social practices that are productive of use value at the same time as they resist the capitalist capture of value.*

As Barron (2013) shows brilliantly, FLOSS can be interpreted as a critical social practice in Boltanski and Chiapello's sense because it is antithetical to property-based capital accumulation, yet "it can be shown to embody the 'new spirit of capitalism' in its most distilled form". Despite the differences, a similar argument can be made for OSHW. Indeed, the management of r0ket production itself have been organised according to the project order identified by Boltanski and Chiapello in *The New Spirit of Capitalism*: a heterogeneous temporary coalition built around an end product rather than a single organisation (2005). In terms of the actual work process in the hackerspace, the nonstop soldering sessions did not look like the New Industrial Revolution professed by Anderson (2014), but rather a happy sweatshop of hobbyists high on caffeine crunching out piece-work in their spare time because manual labour proved to be cheaper, more accessible and apparently more precise than factory production. In this light the situation appears to be a regression rather than a revolution in the history of capitalism. "The idea that any liberation of production must involve an intimate re-engagement with all aspects of it is not liberation, but simply a restoration of an earlier stage of compulsion, with a strong dose of moralism included" writes Rundle (2015), reminding us that since the 1960s numerous community initiatives professed craftsmanship versus industrial production, falling into the trap that they thought alienation can be overcome through the sheer ethical will exercised in small groups. As hinted in the theoretical framework, hackers at the height of their producitivity – like the most intense moments of r0ket production – appear as the ultimate capitalists subjects: working without control, coercion or remuneration only to create innovations that capital can freely reappropriate. Basically, in the worst cases the commons is produced to feed primitive accumulation. However, as I seek to show through both case studies, some of the most interesting artefacts produced in hackerspaces are not sensible or even viable as commodities.

Whereas ideas of liberation-through-craftsmanship are sometimes professed by practitioners, these illusions are actually much more widespread amongst theoreticians who are too quick to extrapolate the progressive tendencies of current practices into a totality that can replace capitalism. The r0ket team had no such ideals, perceiving their role on the level of middle range theory instead, putting into place dynamics that bring participants of their scene closer to each other in general and closer to hardware hacking in particular. Such an attitude is more descriptive of widespread pragmatism that is at least a strong current in hacker cultures. Nonetheless, it is noteworthy that hackerspaces are producing subjectivities accustomed to unalienated labour, subjectivities which have a multi-faceted concept of use value and subjectivities that intrinsically question the division of labour in modern societies. Closer engagement with materials appears to be useful for the acquisition of a critical point of view that in turn informs critical engineering practices. However, in the final analysis it may be a necessary but not sufficient condition for opening up new perspectives to capitalism – or even modernity, for that matter.

9.2.4 The first and second batches of r0kets

The first run of r0kets was based on an extra 15 EURs factored into the ticket prices of the 2011 Chaos Communication Camp, along with vast amount of volunteer labour, added knowledge and sponsorships. The first 3000 attendants of the Camp received their r0ket devices along with their tickets. According to reports 3500 tickets were sold for the camp (Delst 2012) along with perhaps one or two hundred freeriders who hacked themselves through the lax security. Therefore it is safe to assume that the majority of participants were in possession of a r0ket, and the response from the community was overwhelmingly positive, as detailed in the next section.

The experience left the r0ket team equally enthusiastic and exhausted. Therefore they planned to size down and streamline r0ket production for the Chaos Communication Congress. Since the Camp and the Congress are part of the same subcultural circuits it was safe to assume that many of the same people would turn up for both, and the r0ket team urged them to bring their r0kets along for the conference, announcing updates to the firmware and new apps. In order to supply those without r0kets the negotiations started about a second run of r0kets. The three key points of the new agreement are outlined here. Firstly, they could not ask another favour to use a whole pick and place plant for free, and r0ket team members were tired of soldering all those pieces together by hand again anyway. Even though it was a great educational experience the first time around, it was not something they wanted to repeat – and it would not have attracted so many volunteers anyway. They approached Mitch Altman – who already featured prominently in the previous chapters. He is a key figure of the hackerspaces scene as well as fixture of hacker conventions. Similarly to Nick Farr organising the Hackers on a Plane, lately Mitch Altman have been organising hacker tours to China in order to bring hackers and their manufacturers closer to each other and enable the cross-pollination of engineering cultures. With several successful OSHW projects behind him, Mitch new about sourcing. He suggested a company called Etonnet ("Your Strategic Manufacturing Partner") which could help sourcing the parts, print the PCB and perform the pick and place part.[110] The company did not only offer a good price but managed to dispel the image of sweatshops usually associated with the word combination “Made in China”. At some point one of the r0ket team actually visited the company on their premises. Updates have been made to the firmware, but what actually distinguished first and second generation r0kets the most was that Etonnet actually got the colours right, not like the German company before. As a result, the second generation r0kets sported distinctive deep yellow stripes whcih made them easy to recognise. In sum, the OSHW design and FLOSS programming part stayed in Europe while manufacturing have been outsourced to ShanDong.

Secondly, since there was no point in giving out the r0ket along with the tickets to the event like the last time because participants may already have one, it has been decided that the r0ket would be sold at a special booth. The price would be 35 EURs which was only a few cents above the manufacturing price. The Club agreed to provide the seed funding again, provided that there is a guarantee for selling all r0kets. That is where the last deal came into the picture.

Thirdly, a German distributor based in Pförring – an hour drive from München – called Pollin (“Special items at great prices!”) agreed to take the remainder of the second batch provided that they get at least 300 units. Therefore, the plan was to make 1000 r0kets, keep 300 for Pollin and try to sell 700 at the Congress, giving all the leftovers for Pollin. Subsequently Pollin would put them up for sale on their website for 39.95 EURs. The scheme worked out smoothly – 700 units were sold in the space of a few hours — and a year later there were no r0kets stocked by Pollin either.

As a result, r0kets have actually made it to the market as commercial products, commodities to be bought and sold. Taken in the context of the entrepreneurial ethos of the Makerspaces, this could understood as the acme of success for a prototype that have been produced in a shared machine shop. Tuning down from the enthusiasm of makers, it is still a common view between hackers that the market is the most efficient method of distribution for reaching a large number of people. However, r0ket team members displayed a profound disinterest both in the afterlife of their OSHW creation in general and its business potential in particular. Sec pointed out that as OSHW developers, they made sure that everything is available for third parties to adopt the project and take it in another direction, be that profit oriented or not. He was happy with the results of the project and thought that the r0ket made a great development board for hackers to play with and initiate them into the world of electronics, so it reached its goal. Personally, he was happy to move on to learning about something else.

What the second run of r0kets showed however is that since the commodity is the default form of the object in capitalist societies, there is a certain gravity which means that the longer the history of an object, the more likely it will end up on the market. The bias is even stronger in OSHW since the more a

[110] More on these services here: http://www.etonnet.com/service/productionsupport.aspx

project advances the more likely it will need commercial backing. Kopytoff (1986) observed that the "social life of things" passes in and out of the market, and individuation takes place when an object takes on properties or associations with its particular environment – this happens with the r0ket through its distribution at a particular hacker meeting where it becomes the material residue and a kind of souvenir, similarly to the DEF CON badges described above. Even if it is acquired through the market, the fact that it is only meaningful in a particular engineering subculture makes it culturally significant and hence adds to its attraction. The switch from distributing the first run of r0kets to participants of the Camp to selling the r0kets at the Congress and on the website of Pollin can be described as a process of *commodification* according to Boltanski and Chiapello, who theorise that things which were outside of market circulation – and therefore in their definition *authentic* – often acquire exchange value as the history of capitalism advances and it absorbs critiques raised against it. Here we can observe a micro-cycle of commodification where the continued viability of r0ket production depends on its appearance on the market even if the intention of its makers is non-commercial.

While several globally distributed corporations managed to benefit from the project (NXP Semiconductor, Etonnet and Pollin) in one way or another, it is analytically crucial to realise how the relative autonomy of the hacker scene have been articulated throughout building these relationships with the agents of capital. In the most interesting and significant cases during the production of the r0ket device, OSHW Hackers did not meet their commercial partners on the market by simply purchasing their offerings, be they products or services. They approached companies to ask them to do something that was not "business as usual", taking the initiative to shape the context of the deal. They did not engage commercial actors from the basis of the fundamental anonymity that the market provides, neither did they treat these transactions as mere financial operations.

The arguments that r0ket team members used when enrolling participants into their projects pointed beyond commodity relations in the same way that the actual fate of the device itself passed through the sphere of commodity relations only as one phase of its trajectory – the latter which I have already shown in reference to Kopytoff (1986). Studying the history of the reification concept, Vandenberghe (2013) states that "Analyzing capitalism as a system of generalized exchange, Marx notes that the commodity has become the universal form of the product of labor, with the result that the exchange value of the commodity supplants the use value." There are several elements in the biography of the r0ket artefact which subvert both the historical reality contained in that broad statement and the theoretical assumptions used to understand capitalism. While the r0ket itself resists theoretisation as a commodity that represents the universal form of the product of labour, the arguments used by the hackers – for instance in negotiation with manufacturers – cannot be captured by the notion that the exchange value of the commodity supplants its use value. What happened was sometimes the opposite: use value supplanted market value. A factory owner who is by definition capitalist lended fixed capital (a pick and place plant) to the team for free. The capitalist in question decided that the use value of the product is more important than the exchange value he could acquire for pick and place services on the market. Volunteers laboured uncounted hours to make the device a reality because they wanted to use it. Therefore, r0ket team members managed to side-step some of the most binding tenets of political economics. They could be framed as economically rational actors that differentiate themselves by innovative, perhaps even "disruptive" tactics, or social entrepreneurs who make a match between getting ahead and serving the community.

However, in line with the critique of transaction cost economics in the theoretical framework, I would advance another line of argument that is less concerned with the effects or motivations of their actions and more curious about its existential basis. I think what differentiated the r0ket team from other actors on the market is that their strategies were grounded in a unified experience of life: rather than considering the offers of other actors which were advertised on the market, they could look beyond its abstraction to consider the full range of possibilities for action that they or others could take, along with the full range of motivations for those actions. Furthermore, it seems that unlike FLOSS, OSHW still lacks an array of well-established, stabilised patterns on which the relationship between the private sector and the community can be modelled, even though the various forms of sponsorships laid out above start to circumscribe such a tradition. In fact while OSHW advocates are still looking for a consensual definition of their object of desire, Kelty (2013) argues that there is no FLOSS any more in the sense of the historical cultural social formation that inspired so many forms of hackerdom. That is because FLOSS practices integrated to mainstream capital accumulation schemes (also known as business models) and regimes of social control (also known as legal framworks).

9.2.5 Consequences to theory

All in all, we can see that the r0ket project mobilised a set of social relations (sponsorships, collaborations, the Chaos Club) and a technological repertoire (the electronic lab at the hackerspace, IRC and soup.io) which is specific to the hackerspaces, while at the same time (according to the principle of relative autonomy) went beyond the confines of the scene, using proprietary software, Chinese manufacturers and corporate partnerships. These socio-technical choices show the particular challenges that OSHW developers face. At the same time, they underline the argumentation of the late writings of Bauwens in which he offers a post-autonomous conceptualisation of peer production, i.e. that at least in the current historical period, peer production cannot function as a self-enclosed system but have to integrate and interface with other social logics at work in society.[111] The means of how such integration and interfacing happens are

[111] See section 3.3.1.3.

the plane of emergence for the social conflicts that peer production practices engender.

9.3 Integration

The making of the r0ket device should be seen in a series of contribution that µC3 hackers made to the scene. Probably their first major contribution was a flexible electronic system which can control a number of high-power RGB (Red Green Blue – variable colour) LEDs that usually light up sizeable plastic boxes. Huge displays can be easily created from these monster pixels that become the fixture of hacker meetings. On one memorable occasion they were used to light up the glass facade of the Berlin Congress Centre on Alexanderplatz, where for many years the Chaos Computer Congress took place. The installation spelled out the name of the project: ACAB – All Colours Are Beautiful. However, it also conveyed an anti-establishment political message, since as it is well known to many hackers, ACAB is a popular anarchist graffiti that spells out All Cops Are Bastards. Putting the installation in such a prominent place in the middle of Berlin were seen as the confirmation of the relative autonomy of the scene, e.g. that even though hackers meet in an official congress centre in the middle of Europe's veritable capital city, they engage with the establishment on their own terms without forgetting all the repression they received.

But perhaps the most sustained and substantial contribution of µC3 hackers to the regular hacker meetings have been the organisation, coordination and hosting of the Hacker Jeopardy game, played in the central auditorium of the Congress each year. This is also how the general audience of the Congress knows them since some of them are on the main stage while hosting the competition. The original Jeopardy! is a game show on television debuted in 1964 where participants are presented with general knowledge questions that they select according to topic areas and difficulties from a board. The Hacker Jeopardy mimics the stage installation of the show complete with big red buzzers built into the competitor's booths and a software application (Beopardy) that is used for projecting the board itself to the auditorium. Of course the questions are not about presidents of the USA or dinosaurs, but about hacker lore and hacker history. The game — enacted in their native German – enjoys incredible popularity amongst hackers and stays to be one of the highlights of the Congress. In anthropological terms we can say that Hacker Jeopardy is a ritual to reproduce the shared knowledge which is the basis of the subculture. While all subcultures rest on a bank of shared knowledge, since knowledge is central to the identity construction of hackers, it is perhaps even more important for them. Coleman (2012) argues that in-jokes are a constitutive part of hacker culture more than in the case of other subcultures. In-jokes pass on an engineering aesthetics and political taste unique to hackers. As the r0ket device entered circulation in hacker the circuits, it became one of the question categories at the Hacker Jeopardy.

The game is analytically important here as supporting material for my argument that the OSHW presented here – most of which is culturally significant and practically useful only in the context of the hackerspaces milieu – are the products of a particular kind of sociality that characterises the engineering subculture of hackers. A sociality that values cultural and technical contributions on the same level as a human communication. Of course most STS scholars would readily argue that technical contributions are by definition also cultural contributions, but most members of the general public cannot appreciate a piece of code or a refrigerator on the same level as a poem or an opera, and therefore have a hard time recognising it as a piece of culture. Therefore, we can observe that hackers and STS scholars share a common sensibility to material culture.

Similar notions were explored by Jameson's literary criticism of (post-)cyberpunk writer William Gibson's novels – indicatively popular with hackers (Jameson 2003). As Jameson points out, for Gibson a chair is never a chair as such: it is either a "a very long, very low, vaguely ominous and Weimar-looking piece of spring-loaded furniture" or alternatively "a faux-classical Japanese interpretation in black-lacquered wood, upholstered with something that looks like shark-skin" of the same. Similarly, we may add, a surface is not simply smooth but either kevlar or chrome coated. These observations obviously go back to Husserl's phenomenological arguments in *Cartesian Meditations* on intentionality and categorial intuition. He writes that upon seeing a car, we do not see a patch of red and then a car, and then a relation between the car and the red, but we take in the experience of a red car "in a single blow" (1960; Øverenget 1998, 37 41 45 62). For hackers a piece of hardware immediately communicates a given space of possibilities and limitations, and therefore alliance or treason: a set of social relations. If somebody sees an LCD display on the r0ket it can be just an LCD display: but for a hardware hacker it is culturally and practically significant that it is a Nokia 1110, an off-the-shelf part widely available in her life world, suggestive of a society where people can actually repair, rebuild and reinvent their devices, and therefore have a sense of control over their technologies. This evaluation is in stark contrast with consumerist tendencies – and the sociological impression of hackers as *early adopters* – that assume that new technologies are by definition better. It is because of the significance of each component and their instrumentalisation that we have to look closely at artefacts and their social, political and technical dimensions in order to understand how they shape society and how they shape them. The next two sections will do just that: walk through the hardware and the firmware and discern features, putting them in their cultural context. The result is a *close reading* of an artefact in light of the three research questions identified in the beginning of this chapter.

Before doing all that, however, it is possible to give a very simplified answer to the classic question of hacker studies: why they did it? – that is the question of motivation. Overall, I would say that the r0ket device has been produced as an act of love. Love of technology (as Latour would say), and love of the people who love technology: in particular the love of electronic artefacts and the desire to share this love with other people who love technology but perhaps have not been acquainted with electronics artefacts just yet. In this vein the r0ket device exhibits remarkable continuity in the tradition of hacker badges started at DEF CON 14 (2006).

9.4 Hardware: OSHW against stabilisation and closure

As a piece of hardware the r0ket can be positioned on a scale between an Arduino (OSHW microcontroller popular with hackers and artists) and the mobile phone. Gauging the r0ket from a comparative perspective allows for nontechnical readers to understand its functionality in the context of an ever changing technological landscape. In their different ways both the Arduino (Banzi 2008; Paoli and Storni 2011) and the mobile phones are valued for their role in the democratisation of technology, allowing users with less expertise and less money to get connected and integrate into large technological systems. The Arduino provides a user friendly graphical user interface (GUI) and an integrated development environment (IDE) for programming microcontrollers in a high level language (a simplified variant of C called Processing, originally used for real time audio-video synthesis for artistic performances). Similarly to the first hacker conference badges, the Arduino was first available on the market in 2005, thus part of the renewed wave of enthusiasm for physical computing that enabled the rise of the hackerspaces. However popular the Arduino became in the hackerspaces, it does not exactly target the particular engineering aesthetics of hackers because it is more geared towards artists. The departures from the Arduino model that the r0ket team decided on were definitely made to appeal to (old-school) hackers more: the exotic Processing language have been replaced with the industry standard C and refreshing the firmware happens through copying the code to the device as if it was a pendrive, without requiring a special IDE.[112] Once again, we can see that hackers prefer standard and general solutions.

The Nokia 1110 display have been mentioned before along the same lines of argumentation. It is a 96x68 pixel monochrome display, even though both the connector and the firmware supports the Nokia 1600 colour LCD panel allowing users an easy upgrade path. The light sensor above the display is programmed so that the LCD inverts in dark condition, making for better readability.[113] There are four LEDs on the board, one is by default a charge indicator, and two others are called “position LEDs” because they are at the edges of the rocket shaped PCB. They can be controlled from the firmware but serve no special purpose. The last LED is not even turned on from the firmware: it is there for users to find out what to do with it. The main input device of the r0ket is a 5 way button (up, down, left, right, push) that works like a miniature joystick.

The most interesting built-in feature of the r0ket however is its support for radio frequency communication using the OpenBeacon specification. The protocol is somewhat similar to the widely known WiFi (802.11) family – for instance

[112] Technically, the Arduino talks to the computer through a serial-to-USB protocol.

[113] In fact one of the revisions made in the second batch of r0kets was that the light sensor was too close to a LED, so the r0ket picked up its own light.

it uses the same frequency range (2.4Ghz) — but it is radically more simple. According to the OpenBeacon website, the project provides "Active 2.4 GHz RFID Realtime Proximity and Position Tracking using the BlueTooth Low Energy (BLE) compatible nRF51822 chip from Nordic Semiconductors."[114] In practice this means three things. Once, r0kets can communicate with each other across a distance of 6 to 10 meters[115] forming a mesh network to transmit messages across the web of r0kets. Twice, r0kets can detect when two of them are facing each other in close proximity – typically this means two people talking to each other. Thrice, special wireless routers can be deployed to pick up their signals and track the movement and orientation of r0kets in a territory. The architecture has been optimised to consume as little electricity as possible so that simple wireless tags can operate for years from a small round battery. There is support for the encryption of packets so that only designated nodes can interpret them. Exploiting the latest trends in techno-gizmos, the r0ket thus becomes a "social" artefact generating "big data" for "Internet of Things" applications. We shall return to the discussions of these possibilities in the following pages, but I will not engage with the aforementioned hyped up discourses in greater depths. At the moment it is enough to acknowledge that these capabilities are based on a free software protocol stack that uses a commercial chip (Nordic Semiconductors' nRF51822[116]) — a pattern we already identified in OSHW development.

The USB micro-B port serves three purposes. Once, to charge the Lithium Polymer battery.[117] Twice, to access the storage capacity of the r0ket like a pendrive. Thrice, to communicate with the r0ket. This may sound trivial but once again it is a more general solution than what is found in the Arduino because it cannot perform the second function (having no conventional storage device).

In addition to these features, the really useful part from the hacker's point of view are the pin-out connections which allow the r0ket to be part of a larger contraption, e.g. to connect it to other electronic components. There are two possibilities prepared to accommodate these needs. The primary interface of the r0ket is called the *m0dulbus*. The m0dulbus was the component which had to be soldered onto the first batch of r0kets manually, necessitating a collective effort. It is basically a dozen female socket connectors where extension boards can be connected. The latter are called m0duls. A sample m0dul was sold by the r0ket team for 5 EURs. The name of the m0dul was *Flame* and it was comprised of a small PCB with a laser-cutted plastic piece attached: a LED on the PCB filled the plastic piece with light, complementing the shape of the r0ket with an engine flame. The m0dulbus also makes it easy for designers to make

[114] http://get.openbeacon.org/about.html

[115] According to tests I witnessed in the H.A.C.K. hackerspace (Budapest, Hungary).

[116] "The nRF51822 is a powerful, highly flexible multi-protocol SoC ideally suited for Bluetooth® Smart (previously called Bluetooth low energy) and 2.4GHz ultra low-power wireless applications." – from https://www.nordicsemi.com/eng/Products/Bluetooth-Smart-Bluetooth-low-energy/nRF51822

[117] To a maximum capacity of 600 mAh.

stackable extensions which can be combined with each other. The female socket connectors make it easy to add and remove the m0duls from the r0ket. The secondary interface provides an additional dozen pin out points, but they are simple holes on the PCB where the user have to solder the desired wires. Thus the second method is more cumbersome and also less supported in the firmware, but allows for permanent fixtures and maxes out the possible wires to connect to the r0ket to 24 (compared to the Arduino's 20 female socket connectors).

These connectors are also called *breakouts*, because they "break out" the connections from unused legs of a chip. A microcontroller is a chip with a certain number of legs, properly called pins, only some of which are used to achieve the desired functionality of the device. It is almost a rule that whichever device one is looking at, there are always unused pins, because it is rare that the requirements of the implementation match up exactly with the chosen hardware components. However, under normal circumstances the extra legs are left hanging, so to say, unavailable for the user. Hardware hackers sometimes modify defines and "break out" these legs so they can be used for extending the functionality of the original device. In the case of the r0ket and similar hackable artefacts, the original authors of the design include breakouts in their plan, making sure that the users can take advantage of the full functionality of the chips at the heart of the device.

I argue that this gesture is at the heart of eminent OSHW, although it is technically not a requirement for an open source hardware artefact. As already noted, it is enough to publish the schematics, Bill of Parts and documentation for an electronic artefact under an open licence to qualify as OSHW. Therefore, I propose to introduce the concept of unfinished artefacts to refer to electronic devices which are eminent OSHW in the sense that they actively support the social practices associated with OSHW, not merely qualify for its requirements. Furthermore, in the next case studies I will argue that unfinished artefacts are not merely eminent OSHW, but can shed the formal limitations of OSHW. The artefacts discussed in the next case study are rarely licenced as OSHW, yet they embody the OSHW ethos more than most small scale electronic artefacts that are formally considered OSHW. In conclusion, unfinished artefacts are not necessarily OSHW, but they are always supportive of the social practices associated with OSHW, and this aspect is considered more essential than the formal requirements. In conclusion, the conceptual tension between accepted definitions of OSHW and my conceptualisation of unfinished artefacts is useful for bringing out the contradictions in the peer production of OSHW and to enable the analysis of OSHW as a set of social practices rather than a set of legal requirements.

To return to the m0dulbus and Hackerbus: I argue that in the language of SCOT (Social Construction of Technology), these breakouts are functional components whose purpose is none other than to forestall stabilisation and closure, and thus preserve the interpretative flexibility of the artefact. Breakouts are additional physical components that are added to the design in order to provide openness as a feature. This is in stark contrast with the understanding of OSHW as a licencing requirement for the blueprints of the design, because it changes much more than the legal standing of the documentation: it is a modification of the schematics and an addition to the Bill of Parts. In other words, designers who add breakouts to their devices not only safeguard the classic *four freedoms* provided by FLOSS licences to users to (a.) use, (b.) understand (c.) modify and (d.) distribute – they are actively enabling and encouraging these behaviours.

I have emphasised before that keeping the schematics simple and the Bill of Parts short is essential in hardware design for meeting the budget requirements and the deadlines of the project, especially in proportion to the number of units to be produced. Of course in FLOSS the additional costs of introducing new features are mainly measured in the human work time of programmers and it is difficult to estimate how much extra cost an extension of the specification will cost – but in hardware design it can be easily formulated in terms of the market price of additional components and the higher cost of PCBs manufacture. Therefore introducing new functional components to a hardware design warrants a strong justification, which in turn makes it more significant analytically.

When Pinch and Bijker (1984) introduced the idea of closure, stabilisation and interpretative flexibility, laying the foundations to the SCOT approach to STS, the theory came with a number of assumptions. First, the meta-narrative behind the trajectory of technologies described by the adherents of SCOT was a gradual decrease in the interpretative flexibility of technologies, tending from controversy towards consensus. The history of technologies through conceptualisation, development, manufacture, use and obsolescence was generally perceived as a contested ground on which the relevant social groups engage with each other in order to impose their own interpretations about the relevant criteria for judging the functionality, cultural meaning and use cases of the artefact. The closure mechanisms defined in SCOT are the concrete tactics used by the relevant social groups for implementing their strategy. Closure mechanisms are concrete steps towards stabilisation which cement functionality: for instance industrial standards fix parameters which have to be applied in future works, or as Winner (1999) and later Latour (1996) notes the height of a tunnel determines what vehicles can pass under. Closure mechanisms keep some social groups from tampering with devices while at the same time they serve as a guarantee for the reliable functioning of those same devices for other social groups. Therefore they are usually a mixed blessing. Callon (1986) notes that closure mechanisms make actor-networks more durable by blackboxing parts of the network into a single node whose internal structure is not exposed. Plastic boxes covering electronic devices are a classic example of making a device more durable while simultaneously preventing users from tampering with it. Crucially, there is no place for *opening mechanisms* in the theory which would complement the effects of closure mechanisms. The explanation is that it has been assumed from the start that each relevant social group is engaging with the development of the artefact in order to drive it towards stabilisation according their technological frame (i.e. interpretation).

Second, another assumption was that stabilisation is a necessary but not sufficient condition for getting something that “works” at the end of the process. Since each relevant social group has their own conception of the artefact in general and what it should do in particular, the one that wins gets their own version of the artefact or at least the discrete functionality they were struggling for. Projects that failed to produce a stabilised artefact are most likely failures as a whole because the negotiations stalled at a point where the technology could not be developed further, and the device dies a premature death.

In my interpretation hardware hackers and the opening mechanisms they advocate appear as an anomaly in the classic SCOT model that enables us to further refine the theory while understanding this particular empirical case in more depths (Maxigas 2014a; Aibar and Maxigas 2014a). Stabilisation is challenged by hackers in a number of ways, which can be divided into four broad categories. First, through social practices of knowledge sharing and technical experimentation that question the social conditions necessary for the stabilisation of the artefact. Reverse engineering when a proprietary device or protocol that is supposed to be a black box for its users is discovered through technical interrogation is the most typical of this first category. The aforementioned XBOX hacks are great examples of users understanding their devices and extending their functionality despite all the technical, legal, and moral countermeasures by the vendor. I propose to call the resulting artefacts *opened hardware* because they are not technically open source, yet adequate documentation circulates about them to enable users to exercise the four freedoms associated with open technologies, i.e. to use, understand, modify and distribute them.

Second, hackers challenge stabilisation through legal instruments such as copyleft licences which define the right of users in the licencing agreement. As widely noted by theorists and practitioners alike, copyleft licences subvert the original intention behind the establishment of licencing, since the original intention was to restrict kind of things that users can do with the technologies at hand. FLOSS and OSHW is defined through its licencing schema, and hacker technologies are widely identifies through their legal status by scholars and practitioners alike. Since the legal issues around FLOSS/OSHW along with their social, political and technical consequences is a widely studies area, I will not focus on these aspects here. Rather, the main thrust of my research project is to broaden the understanding of FLOSS/OSHW and the technological repertoire of hackers towards a set of social practices and social relations that go beyond the question and consequences of licencing. I am proposing the term unfinished artefacts exactly because such a broad understanding goes beyond the established realm of FLOSS/OSHW.

Third, what we can learn from the m0dulbus and the Hackerbus is that beyond social practices and legal instruments there are also specific technical features through which hackers challenge stabilisation and closure. All three aspects (social, legal, technical) tend to occur together since they are analytically distinct parts of the same engineering subculture, but any of them could work alone and could be effective as an opening mechanism. Breakouts provide an interface to the device that makes it easier for users to modify it and use it as a component in a larger technological system of their choosing. Even if the r0ket would be legally established as a proprietary device and the developers would guard their knowledge of how it works, the breakouts would make it fairly trivial for users to exercise the four freedoms associated with OSHW. All the modifications that are listed in the section of this chapter on usages would still be possible and would probably happen despite the social and legal resistance of the r0ket producers, because it is enabled through the very functionality of the device.

To summarise: the breakouts on the r0ket are technical elements that allow users to shape and extend the functionality of the device, thereby forestalling stabilisation and closure. In this capacity they can be called opening mechanisms – a notion that complicates the SCOT understanding of technological trajectories as trending towards stabilisation and closure, while questioning the second assumption that functional devices are by definition stable devices. In fact the breakouts add functionality to the r0ket, and destabilise the whole composition of the device, without actually crippling any other functional elements in the ensemble. Needless to say, there are many other aspects of the r0ket that make it hackable, and the breakouts are only one example amongst many. I choose to focus on it because it is the most prominent feature and it is easier to explain than some other more esoteric functionality. In the larger scheme of things hackers appear to be a social group whose participation, initiative and intervention in technological trajectories tend to oppose stabilisation and closure through knowledge sharing, legal instruments and engineering work. *Openness is introduced to the r0ket not simply as a licencing scheme or a design principle of reproducabilty, but an active intervention into the functional composition of the artefact: it is introduced as a functional feature – a discrete technical component.*

What is even more interesting, however, is that hackers are a relevant social group which is thematising the very topic of stabilisation and closure in their discourses and practices. Through questioning the host of social practices, legal instruments and technical features which can prevent users from tampering with their devices, they are translating broader social issues to practices, licences and design norms, such as the ones described above. Their engineering aesthetics promotes functionally robust devices that at the same time maximise their interpretative flexibility so that users can modify the trajectory of the technology and take it into different directions. Having said that, it is important to add that hegemonic design practices in the field of ICTs already exhibit some of these aspects, even if in a milder form. While planned obsolescence is still a widespread design goal in commodity electronics, in the fast moving ICT market there are also mainstream tendencies that seek to create technologies for longevity: ones that can adapt or can be adapted to the changing requirements of the field.

Moreover, the observations about opening mechanisms do not invalidate the classic cornerstones of SCOT theory in any way. As we have already seen, when hackers engage in a struggle against closure and stabilisation, they do so using the powerful closure mechanisms of industrial standards, and they stick to them even more than mainstream engineers do. Instead of designing an innovative connector, they insist that the pins of the breakout on the m0dulbus and the Hackerbus have to be the standard size and shape established by previous electronic products. If something was made to interface with the Arduino using its female socket connectors, users should be able to connect it to the r0ket in the same way. Since the main thrust of the design goals is empowering users and expanding the interpretative flexibility of devices, the designers of the r0ket do not see the Arduino as a competitor in the same way that a rival corporation would do. Therefore there is no incentive to introduce any means that would artificially make the r0ket incompatible with the Arduino — in fact the use cases for the r0ket presented two sections down testify that users often combine the r0ket with the Arduino, which happily expands the possibilities of both devices. The only way in which the r0ket competes with the Arduino is that it tried to target a somewhat different audience with a different design which works better for certain use cases.

In conclusion, opening mechanisms themselves, as well as other design tactics to ward off stabilisation and closure tend to stabilise themselves into standards or best practices. Similarly to the legal tactics described many times in conjunction with copyleft licences, hackers paradoxically use these stabilised opening mechanisms like breakouts as closure mechanisms which restrict the interpretative flexibility of the device in a specific way, i.e. to prevent closure and stabilisation in the sense of locking down the device and its functionality. Therefore anti-stabilisation tactics also stabilise to become standards and best practices, even though they never become black boxes: everybody should know how they work and how to modify them. These technical measures are one aspect in which unfinished artefacts can go further than standard OSHW on the path of openness.

The last hardware aspect of the r0ket to be introduced in this section is a great representation of such tendencies: the r0ket comes as a naked BCP without any kind of case. When the user holds the device in her hands, all the electric circuits are laid out clearly in front of her eyes, and all the electric components are exposed. Therefore, it is much easier to see how everything works and what to expect from the r0ket. Even if one does not immediately grasp the technical composition of the device, the exposition of its internals helps asking questions and answering them didactically. There is no black box to open in the case of the r0ket, because there is no box: the design rhetorics of the artefact invites destabilisation, reinvention, and ultimately: hacking.

However, unfinished artefacts are not unfinishable. Exactly because of their high interpretative flexibility that allows users to develop them further in different directions, users could choose to develop them in the direction of closure and stabilisation by removing those safeguards. For instance a major corporation in the event logistics business could take the r0ket, modify the shape of the PCB, strip the hardware of breakouts and the software of crazy applications, and get a device that is only good as a name tag that can exchange electronic business cards wirelessly between participants. It would only take a few days' work and they could offer it for a 60 EUR extra fee per head for the major industry expositions as a novelty option. The company could make a profit on the side by tracking people at the conference and selling the metadata to the organisers. Of course that would not prevent the OSHW hackers to continue using the r0kets and making new generations of them according to their liking, but it would intervene in the trajectory of the technology in a meaningful way. As in FLOSS, the commodification of OSHW also happens throughout generations of technologies, via upgrades and "improvements" – a story we have seen with the RepRap printer versus MakerBot Industries.

9.5 Software: Material condensation of a cultural microcosm

The tour of the hardware sought to situate the r0ket in the wider technological landscape and simultaneously flesh out the concept of OSHW, as well as introducing the idea of unfinished artefacts. Going through the software features of the r0ket on the other hand is an opportunity to turn our attention inward through the diversity of the hacker cultures cultivated in the hackerspaces milieu and at the hacker conventions where the r0ket has been distributed. I argued that the r0ket team acted as impromptu anthropologists when they set out to create an electronic name tag which most hackers like. In their attempt to appeal to the cultural and technological taste of their audience they performed one of the most important jobs of an anthropologist: to gather into one place all the disparate elements that make up hacker culture.

The r0ket comes with a firmware that displays the configured nickname on the display, presents the user with a menu on the click of the joystick for reaching more functionality, and provides a framework for modifying the software through the USB cable. In this section the features of the default firmware are described, while some of the later modifications will be mentioned in the next section under *Use cases.* The menu contains 5 options. (1) The configuration menu. (2) An "execute" menu for launching programs. (3) The "messages" that come from the mesh network. (4) "Nick" sets the nickname shown on the badge and its font and background animation. (5) The last option simply turns the device into USB storage mode, used for data transfer. The more interesting functionality comes with the programs included on board called l0dables. There are seven of these pre-loaded on the r0ket.

BLINK simply blinks the red led above the display. This is a trivial program known from the Arduino development tradition which serves as the basis for newcomers to develop more sophisticated applications, show how to reach system resources and test whether the hardware of a new r0ket is working properly. I already noted the connection to *hello world* programs — it is safe to assume that most programmers' "first word" in a new language was "hello world" (printed on the screen of course). Similarly, most hardware hackers start out by blinking LEDs and then move on to more sophisticated applications. The fact that this is one of the few pre-loaded applications can be interpreted as an invitation for incremental development. Even if users don't have access to the documentation, they can connect the r0ket to their computer, switch on the USB storage mode, and see this little program on the device. Including the BLINK program is a gesture of unfinished artefacts: the program is not incredibly useful in itself, but it is simple, standard and transparent to the user. Therefore, rather than a finished feature, it is a meta-feature inviting the user to extend the functionality of the r0ket – in other words, to test the limits of its interpretative flexibility. People who like to do this are called "developers" in the hacker scene, but there are many other kinds of hackers, even at a convention like the Chaos Camp or Congress.

INVADERS is an implementation of the most iconic single player game ever invented. It is so well known that the reader probably already knows the game-play. The (post-)cyberpunk author William Gibson invented the word *cyberspace* in reference to the conceptual place occupied by teenagers playing such classic games in an arcade hall. The 5-way joystick on the r0ket is ideal for playing this game. Old-school (8 bit) games in particular are overly popular with hackers, so much so that refurbished arcades stand in many hackerspaces and no hacker convention is complete without a Retro Gaming Area where they should run on their original hardware. Similarly to the fetish of plain text interfaces (terminals), hackers value the minimalist aesthetics of these early games and argue that the game-play and symbolic universe of a game is much more important than its special effects or realistic graphics. Many of these games are *abandonware*: even if they are not technically free, the owners of the relevant copyright or trademarks are unlikely to enforce them any more. As *orphan works*, they also fall into the category of opened source software. Even if the company did not publish the original source, armed with today's research tools it is comparatively easy to figure out how these old games work.

It is interesting to see that the ideas of simplicity, elegance and standards which guide the hand of hackers when they design a technology, are also cultivated in their purely aesthetic preferences of which computer games they like. Moreover, parties at hackerspaces and conventions routinely feature a genre of music called chip tune that recycles the aesthetic of 8 bit computer music. "Orthodox" chip tune uses only the original chips (like the SID audio chip of the Commodore 64 or the more sophisticated capabilities of the Nintendo DS, the best selling hand-held console of all time), but other types of chip tune music may use the full scale of possibilities available to contemporary music-makers in a way that mimics those "authentic" sounds. All these preferences in engineering, games and music go back to the idea that the application of simple rules can produce complex results and that a "hacker is someone who experiments with the limitations of systems for intellectual curiosity." (Schneier 2000) Simple systems which are cleverly designed are ideal playgrounds for the imagination of hackers. The r0ket itself is one: while it can do more than an Arduino, its capabilities are certainly below any mobile computing devices even by 2010 standards, yet its flexibility both on the level of the hardware and the level of the firmware goes a long way.

MANDEL is a Mandelbrot fractal viewer, which displays the image of a Mandelbrot set on the screen and the user can span or zoom in and out using the joystick. The author of this application said that it was the first serious program to be written for the r0ket and he thought it is worth to include it because it showcases the superior speed of the r0ket's Cortex-M3 processor. However, this choice also highlights the connection of hackers to more obscure areas of mathematics, especially recursion which is the fixation of many hackers and the source of infinite in-jokes in the community. Such "demos" or display hacks were used on early computers to test and explore the capabilities of the equipment. Experimenters sometimes discovered bugs or unpredictable behaviour which could be used to create graphics effects deemed otherwise impossible on the given hardware.

In a sense BLINK and MANDEL are two ends of a spectrum: while BLINK is the most simple test possible, MANDEL is still a test but a much more sophisticated one which pushes the limits of the hardware. Neither of these programs go beyond the world of engineering to do something that mere mortals consider useful. In a way these programs were written by programmers for programmers.

MANDEL introduces the sensibilities of the *demoscene* to the world of the r0ket. The demoscene is one of many hacker cultures that hackerspaces host, and one of the oldest at that (Carlsson 2009). Most active in the 1990s, it centred around teenagers writing demos: procedurally generated non-interactive audio-visual programs rendered in real time. The cinematic aesthetics of these *productions* was greatly influenced by the video clips that were also at the height of their popularity at the time, while their engineering bravado was necessitated by the technical limitations of contemporary home computing: the ZX Spectrum, the Commodore 64 and the Amiga. *Sceners* form *groups* which meet at the demoscene *parties* to show off their *demos* to each other in *compos* (Polgár Tamás 2005). The part for sceners is like a convention for hackers – a social meeting where they present their work to each other. The website Pouet is the canonical repository for demoscene productions.

The crucial point here about the demoscene is the criteria for judging demos: a good demo is aesthetically attractive while technical virtuoso (Tasajärvi 2004). Therefore, categories of demos are defined according to the technical limitations in place, which are most often the architecture of execution and the size of the executable. For example a 4K have to fit into 4 kilobytes (32768 digits of one or zero) or a C64 demo have to be executed on a Commodore 64 computer. Lindsay (2003) shows that long after the commercial obsolescence of a product and disappearance of the vendor, users of old computers can continue to take on the roles of designers, producers, marketers, distributors and technical support for the machine. While she deals with individual home users in her study, sceners continue to push the limitations of architectures collectively. For instance the big news of the demoscene at the time of writing (early 2015) is the technical breakthrough in a new demo entitled 8088 MPH which uses the four colour CGA monitor to display 1024 colours through clever tricks that rely on a detailed understanding of the hardware (Trixster 2015; VileR 2015). If vendors of the time knew how to do this then they could have marketed this monitor as a colour monitor right from its appearance in 1981. However, it took amateurs 34 years later to invent the proper rendering method (*mode*). Since old architectures are so well understood today, it is increasingly hard to produce such virtuoso performances on them. As the makers of 8088 MPH state, it is a

> demonstration in 2015 for the original IBM PC and its Color Graphics Adapter. This demo is the result of two man-years of hard work, and decades of study towards unlocking this platform's secrets. We hope you enjoyed watching it as we enjoyed making it. [...] This is only the beginning (Hornet, CRTC, and Desire 2015).

Hackaday commentator Brian Benchoff notes that (2015), "[a]rtists have been working on these old machines for decades now, and every single ounce of processing power and software trickery has been squeezed out of these CPUs." Therefore, contemporary demo sceners moved on to produce primitive hardware especially for making demos not unlike the r0ket (Benchoff 2015) or hunt devices with limited resources like oscilloscopes (Fabio 2014) and photocopiers (Gordon 2014) where the first demo can be made and their technical possibilities explored (Heikkilä 2010). MANDEL can be put in the same stream as an impressively fast rendering of fractal zoom on a relatively unexplored architecture and the demonstration of infinite resolution on a 96x68 pixel display. The dual technical/aesthetic sensibilities of sceners highlight the close connection in hacker culture between engineering practices and the cultivation of a common taste. I already argued that peer production practices blend education, research and manufacturing in the context of an informal sociality that is technologically productive. Here, the culturally enforced division between people with STEM strength and people with artistic talents is questioned. Further on I argue that the subversive edge of hacker culture comes from an unified vision of life that vanishes when engineering is instrumentalised.

Westcott (2012) is a participant of the aforementioned demo scene: he was coming from a demo party in Helsinki, Finland (Assembly) and going to a demo party in Cologne, Germany (Evoke). He only attended the first two days of the Chaos Communication Camp, where he got a r0ket with his ticket. Three days later he won fourth place in the demo competition in Cologne, while the Camp was winding down. His entry was *wolfy: Wolfenstein for the r0ket badge*, a raycasting engine that renders quasi three dimensional environments on the tiny screen of the r0ket. Since demo scene producers specialise in real time audiovisual synthesis on limited resource systems, for them a three dimensional rendering engine is an important achievement in pushing the limits of what can be done on a particular platform. For peer production theorists, what is interesting in Westcott's hack for the r0ket is that his little program will probably never be used – indeed, four years later it is still a stub.[118] He did not write it to solve a practical task for end users. He wrote it as a particular form of social interaction with his scener peers: software developers who appreciate a good program for what it is, rather than for what it does. For the same reason, he did not need to worry about protecting it with a licence from evil corporations. The source code (289 lines written in C) is not explicitly licenced, which means that it falls under default copyright laws, but since it is published online, everybody can use it all the same. Once again, the social life of this hack is deeply reminiscent of the open source ethos, yet it is not the licencing any more which makes it part of the FLOSS world.

PWGEN is a simple password generator, mimicking the pwgen utility available on most Linux distributions. It generates 8 character long passwords from the 94 printable characters of the most primitive ASCII encoding table (the 95th — space (!) — is not included). PWGEN supposedly generates passwords as random as possible so there would be no logic behind them to guess. However, it has been revealed by the author of the program during the Chaos Communication Camp that this password generator is intentionally flowed! Its algorithm only generates 65536 unique passwords instead of the expected 6095689385410816. The announcement was made anonymously on Pastebin (guest on Pastebin, August 14th, 2011), but the link was included in a post on the r0ket soup of a file with the list of all the passwords from PWGEN signed by mazzoo (Matthias Wenzel).

There are many twists and in-jokes in this story, as the title of the pasted manifesto — "there's no security in trusted boot - or - how I hacked 3000 hackers ;)" — suggests. The way of publication is peculiar to hackers. Although Pastebin is legitimate business helping users to share snippets of text and source code with each other, it is also used by various hacker groups like Anonymous for "releases". A release is a batch of stolen data from a server that was compromised by the poster. Releases usually have a kind of foreword in which the hackers explain the purpose of the attack, the moral of the story, and boast about their own skills. Of course, the release usually targets an enemy rather than one's own group, or at least a splinter cell, but here the one r0ket team member playfully compromised the development efforts of the r0ket itself. The particular reason was to protest the decision that first generation r0kets with the default firmware could only run executables signed by the r0ket team, which was a limitation on its use and thought to be a security measure. However, this security measure is very similar to the anti-hacker technology used by big corporations called DRM or Digital Rights Management. DRM is built into systems to disable people from modifying the functionality of the device and therefore opposed by most hackers. The security argument for DRM is to make sure that the programming code is executed in a well-defined environment which is supposed to be more predictable. One argument against it — recounted by mazzoo in the manifesto — is that even if the source code is available and its readers find problems (bugs) in it,

[118] See https://github.com/gasman/wolfy

they cannot fix the problem themselves but have to go through the manufacturer of the device to acquire a signature before they can run their perfected code. Therefore, the fix is delayed and depends on the "good will" of the party who has the power to sign the new code. By slipping a security hole into the source code of the default firmware on the r0ket the author called attention to this problem and encouraged the r0ket team to remove the limitation, which they indeed did in the second generation firmware prepared for the Chaos Communication Congress. This is what "no security in trusted boot" means in the title of the manifesto.

The general reason for the release, on the other hand, was to target the audience of the r0ket and call attention to the fact that none of the users reviewed the source code of this little security sensitive application before running it, or at least if anybody found the bug they did not publish it before its author did. The moral of the story from this point of view is that you cannot trust any source code that you have not verified to make sure that it does the correct computations. The concrete vulnerability created by this bug is that given the list of the few possible passwords it is exponentially easier to look for machines on the network that use these passwords. This is what "how I hacked 3000 hackers" means in the title of the manifesto. Of course this is mostly theoretical because as mazzoo also points out there is little chance that these 8 character passwords would be used for anything serious. Ultimately, this hack was only an example of the hide and seek hackers like to play with anybody, even with each other, and adds another aspect to the sources of hacker subculture outlined by these applications: there was the demoscene and old skool gamers, and now (grey hat) information security research.

Once again, even though the program was not performative in the technical sense of generating good passwords, yet it was performative in the social sense of having a moral and raising an issue that got fixed the next time the r0ket firmware was distributed. Interestingly, the issue itself have been framed as a technical matter of improper security measures. However, *its moral was that without giving away the power to users of understanding, modifying and distributing the technology in their hands there will never be proper security.* Therefore, from an STS point of view we can say that the PWGEN saga championed the democratisation of technology.

ROCKETS is similar to the "messages" option, except that it merely prints out the nicknames that are broadcasted by nearby r0ket devices. Such simple functionality still plays on the specifics of the hacker community, since — even more than in any other social group — it can easily happen during a hacker gathering that you are face to face with your online acquaintance without recognising her. Indeed, attendants often say that hacker conventions are useful for putting face to names. This is why it is especially useful for hackers to know the nicknames of the persons around them, not to miss any chance to meet a friend or idol. Once again, hackers would traditionally guard personally identifiable information to the point of paranoia, especially if it links their online identities to their offline persona. Therefore it is unlikely that they would advertise their nicknames at an airport or a shopping mall - it takes the community atmosphere of the hacker convention for them to open up to their imagined peers. Of course, the r0ket would be a Trojan horse if it would indiscriminately spy on its users – therefore there is a privacy setting in the configuration menu where 0 means nothing is signalled, 1 only location and 2 sends the nickname too. Compared to the other apps, this one seems to be at least theoretically useful, at least in the specific situational context of hacker conventions. However, like the others, its thrust is to enhance a particular sociality that is based on the exchange of clever technological artefacts, where peer production is but a medium of socialisation.

The last two applications, RECVCARD and SENDCARD, are used for exchanging electronic virtual business cards between two parties using the standard vcard format. As the reader would expect, not many hackers have real business cards. In other subcultural groups this is not really a problem since people usually only need to know the name and email address of each other. However, with hackers it is more complicated. Even novice hackers know how to spoof (or forge) email addresses exploiting the fact that email is one of the oldest protocols that was designed in times when the few people on the network actually trusted each other. Moreover, due to their work hackers are more paranoid about the organs of the state and capital – or their fellow hackers – capturing their messages and listening in on the conversation. Therefore they use strong encryption tools like Pretty Good Privacy to sign and encrypt their emails. This requires the exchange of electronic fingerprints (16 hexadecimal numbers) in person, which ideally should not be done through a computer and a network, but through some other means. One such is provided by the functionality of the r0ket to exchange electronic business cards. At the end of day, the r0ket owner can simply download the gathered business cards to his or her computer as if from a pen drive. The theoretical conclusions we can draw from this application largely overlap the results obtained from the previous one. In terms of communities, this feature addresses the more worldly side of hacker cultures. The sad reality is that instead of pushing the limits of systems, doing deep mathematics, or infosec research, most hackers work as web developers, data scientists or other classes of symbol analysts, whose main problem domain is retrieving, parsing and passing messages in various formats. Transmitting virtual business cards touches on that problem domain and therefore qualifies as a "real world" application on the r0ket.

The tour of the default applications (called l0adables) on the r0ket was destined to show their cultural context, explaining why certain types of hackers can find each of them inspiring. The close reading of the software suit was a good apropos for a panorama of the hacker cultures that meet on the site of the hackerspaces scene, and why small scale eletronic artefacts like the r0ket are interesting for them. Taken as an eminent archaeological artefact suggestive of a certain precisely delimited culture, the r0ket badge can be seen as the material condensation of the hackerspaces scene, a microcosm of hacker culture. Furthermore, on a higher level of abstraction, the very gesture of gathering a colourful culture into a single material artefact is a distillation of the essence of hacker conventions, which are themselves a material consensation of hacker cultures. The r0ket as an unfinished artefact that you can hold in your hand is all the more expressive amidst the 2011 edition of the Chaos Communication Camp as an unfinished architecture that facilitates such peer production practices.

In line with the conclusions of the previous section on hardware where I argued that eminent OSHW such as unfinished artefacts seek to preserve their interpretative flexibility beyond the design phase, here I argued that most of the software shipped with the r0ket is but a demonstration of the r0ket's technical capabilities, and thus little more than an invitation to explore its potential further. Since the r0ket was made to initiate hackers to the world of small scale electronic artefacts and hardware hacking, it is no wonder that these applications are not addressing practical problems that normal users encounter. Nonetheless, they are as efficient as they are functional and reliable. The upcoming use cases section shows how users appropriated the r0ket, answering to the gestures of invitation inscribed in the unfinished artefact, welcoming users to exploit its hardware and software features in unexpected ways.

9.6 Use cases

The most used l0adable contributed to the r0ket badge was the *fahrplan*[119], which means timetable in German and that is how the Camp and Congress schedule is traditionally called. Attendants could access the whole programme (211 events during four days — 52 events a day) through their conference badges. The program showed the currently running events when it started. Work was done to detect the location of the r0ket within the venue and show the event going on in the closest location, but as far as I could confirm, the latter feature has never been implemented.

The second most popular extension was a hardware hack that was developed by r0ket team and µC3 member kiu (Simon Schoar) to provide an example of the m0dules[120] – additional PCBs that could be connected to the m0dulbus sockets. It is comprised of a few electronic and an acrylic body which is illuminated by a relatively powerful LED. The flame came in several editions with differentiated by the laser-cutted logo on the acrylic. The logos themselves continued the encyclopedia of hacker culture hinted to in the previous section, from space invader characters through the logo of the µC3 hackerspace to "Don't panic" – a quotation from the Hitchhiker's Guide to the Galaxy (Adams 1985), a favourite sci-fi parody reading of hackers. This was sold in the µC3 tent during the Camp and when the team set up shop at the Congress they offered it for 10 EUR each (the other extra offer was colour LCD for 5 EUR). It came in a kit form: a little plastic bag of electronic components accompanied by an assembly manual. The

[119] Developed by a user called r0y and committed to the official r0ket repository: https://github.com/r0ket/r0ket/blob/master/firmware/l0dable/fahrplan.c

[120] These are called shields on more conventional development boards.

r0ket team was quite easy to find in both events and its members spent countless hours helping people solder together their flames.

Since the r0ket comes as a naked PCB without a case, many hackers were inspired to make custom cases for their devices. Knitting circles are regular fixtures at hacker conventions, and r0ket cases were one of the most popular things to knit. Once again, everyone had a new r0ket and those who wanted to learn or practice knitting could finish a case in an hour or so – and continue their way with their new product hanging around their neck, displaying their skills and protecting their conference badge. Many others used 3D printers for making plastic cases – these designs are still available in repositories and easily reproducible. A complementary hack was to use the top of a classic ball point pen as a clap-on button for the joystick. Since cases could be made from a wide set of materials, virtually any machine tool around could be used to invent and made new kind of customised r0ket cases.

Stepping further on the scale of typical hackerspace technologies, two r0kets were used to control quadcopters remotely: one sitting on the copter itself controlling rotor motion according to the signals sent by the other that function as a remote controller. The hack showcased the versatility of the r0ket, since the faster ARM Cortex CPU was actually a better fit for the job of motor-control than the usual Arduino, and the 5 way joystick, LCD display and wireless figured as a ready-made remote control. Similarly to aeroplanes, quadcopters are typically equipped with position LEDs — in fact, for safety reasons it was banned to fly any drone at the Camp and Congress without lights. Therefore this hack took advantage of the build-in LEDs of the r0ket too. The double-r0ket hack has been expanded by the same people (whom I only managed to observe from a distance) to a three-r0ket hack where they constructed a remote control car. Two axles of a miniature 4x4-wheel drive were driven by a pair of r0kets, a third used for wireless control. Rapid prototyping boards are generally used to connect sensors and actuators together according to some logic, but the r0ket went beyond this concept and included many of the most popular things that users normally hook up to such a microcontroller. These elements in place, prototyping became even faster, which was an even more important factor in the action packed atmosphere of hacker conventions where time is in short supply.

Jeff Keyzer arrived to the Chaos Camp with an almost complete OSHW Geiger counter kit that he built in the wake of the Fukushima Daiichi nuclear disaster (March 2011). Kera, Rod, and Peterova (2013) in their article Post-Apocalyptic Citizenship and Humanitarian Hardware relate that a similar OSHW Geiger counter kit has been developed by members of a Japanese hackerspace and used by citizens for independent verification of the official measurements reports. He gave workshops each day for people who wanted to learn how Geiger counters work and helped them to assemble the kits he was selling (Keyzer 2011). However, his model only beeped and flashed in response to the radiation levels. Combining it with the r0ket allowed displaying the exact radiation levels on the screen as well as storing and transmitting them for publication and cross-examination with the results from other stations. Better still, both devices operate on batteries, so they can easily by combined into an autonomous off-grid station in disaster areas. A l0adable supporting Geiger counter connections have made it to the default firmware for the second batch of r0kets released for the Congress a few month later,[121] which made it even easier for workshop participants to use the two devices together (anonymous 2011). A similar hack was to use the r0ket as a primitive oscillator, so that it displays the resistance between two connected conductives (like your fingers) on the LCD screen. Oscillators are essential for a great number of basic work processes in open source hardware, like debugging faulty connections on r0kets, for instance.

In a Lightning Talk during the Chaos Communication Congress Tobias Weyand and Christian Buck launched okr0ket, a locative dating application for the r0ket (2015). Like most dating applications, it involves answering a series of questions to map the characteristics of users. Unlike most dating applications, it does not store the answers – which are considered personal and sensitive information – in a centralised database, but broadcasts them to nearby r0kets. A partial match is displayed on the screen as a kind of pop-up window, and a best match is signalled by additional flashing LEDs. The authors claim to have achieved a reasonable accuracy when tested against existing couples. Again, the questions read like yet another mini-encyclopedia of hacker culture,[122] from the significance of mystical numbers like 23 or 42[123] to what is called the "religious wars" between vi (the editor of the devil) and emacs (the first Free Software application to bare the licence).

There is much to be said about the appropriation of genres into a culture here. Weyand and Buck did not try to write the best dating application there is, or even a better one than the already existing dating applications. okr0ket is not a prototype in the makerspace sense of intended as being a crippled precursor for a potentially successful commercial product that can be shows to potential users and potential investors. Lastly, it is not even a symbol object created in protest against the state of the industry, although there is much to be made from that reading and we will return to it. okr0ket is a gift that does three things. Once, it is a sketchy representation of a culture which is not complete or accurate but it is nonetheless recognisable. Hackers see themselves and the things that are important for them in this application. Its primary characteristic is that it was meant for a particular type of person, the member of a relevant social group. Twice, it is an invention that is aimed at helping hackers find love. Its secondary characteristic is that its purpose is to make the members of this relevant social group happy. Thrice, it is FLOSS so that its users are free to use, study, modify, develop or degrade, let alone distribute it. Its tertiary characteristic is that it transfers its ownership to its target audience. These three points more or less define the colloquial meaning of a gift. If I receive a gift, I expect to recognise

[121] The Camp is in August and the Congress in December.

[122] https://github.com/0xtob/r0ketstuff/blob/master/datingquestions.txt

[123] By now the reader should be familiar with the occult significance of these numbers, but for the sake of clarity, the source are Shea and Wilson (1984) and Adams (1985)

that it is for me particularly and not for others, I expect that it should bring me joy, and I expect that I can keep it. The corollary is that I may just put it on the shelf and remember it fondly in sentimental moments, and this is exactly what happened to the okr0ket application: after a five minute presentation that received a raving response, it has not seen much use or development since then, and by now it largely fell into obscurity. Yet, I describe it here because I think it helps to understand the basic human gesture behind unfinished artefacts, a gesture that can also be found in many FLOSS and OSHW projects but does not define them as such. Notably, there is nothing technological about such a gesture in general. Nonetheless, the okr0ket in particular is a technological artefact because both its practical creation and theoretical appreciation necessitates some level of technological expertise and experience. It entertains an ironic relation to the project of the democratisation of science and technology because it is an innovation that only empowers those who could have invented it anyway.

While the games entertain individuals, the dating application brings together couples, there were experiments at the affective aggregating larger masses too. The r0ket team presented a live demo during their r0ket talk at the Congress which let the audience play the Pong game collectively, using the r0kets as a remote control. The Pong screen was projected on the stage while the right side of the audience controlled the right pad, and the left side the left. The MP0NG application allowed participants to join the game and then they used the joystick of the r0ket to make the pads go up and down. The pads moved according to the aggregated will of the relevant side of the audience. The performance went smoothly and solicited an enthusiastic response from the audience.

The show had an important historical antecedent that Curtis (2011) documented, interpreted and popularised in a documentary series released in May on the early history of computers and networks that figured in the documentary as an analysis of cybernetics as an all-encompassing ideology. The specific experiment was conducted by Californian entrepreneur Loren Carpenter in 1991. The audience sat at a large auditorium looking at a cinema screen and each participant was furnished with a ping-pong paddle. The two sides of the ping-pong paddle were painted in different colours. No instructions have been given, but it slowly dawned on them that they are in control of the Pong game projected on the screen. If all people sitting on the right shows up the blue side of the paddle, the pad on the right goes to the top. Conversely, if all people sitting on the left shows up the red side of the paddle, the pad on the left goes to the bottom. After the rules dawned on the audience they were able to play the game reasonably well, and enjoying it too. As the interviews and interpretations made by Curtis (2011) show, the collective Pong game was organised and understood as a laboratory social experiment to prove that self-organising systems can exist in society given the right technology to provide the feedback loops. The experiment was a perfect metaphor for the efficiency of the invisible hand of the market and the viability of an anarchist society, and was used to justify both. The (paradoxical) claim was that technology can be used to enable the emergence of an inherent order in society without any state control, representation or intermediaries, kings and leaders. As in most laboratory experiments, the visible hands of those who designed, implemented and operated the technology in a particular way was taken out of the equation. Thus the experiment demonstrated not only how the invisible hand of the market works but also how commodify fetishism masks the human labour of (social) engineers.

Indeed, the most important difference between the Californian experiment and its European restaging was that in the latter case the engineers were on the stage. The game was a live demo of the technological potential of the r0ket following a half an hour lecture on the details of the implementation and the process of development. Therefore participants had a basic understanding of what was happening and since the r0ket is OSHW, the possibility to take the results and organise the experiment in a different way. The r0ket team itself used the code base to develop a mass voting solution for the Hacker Jeopardy mentioned above, where the audience could decide whether the solution to the riddle uttered by the on-stage competitors have been correct.

Since the r0kets could talk to any device with an antenna tuned to their frequency and a protocol stack implementing OpenBeacon, the r0ket team placed more stationary and high-powered equipment on the territory of the Camp and Congress, broadcasting such things as the current high score of the SPACE INVADERS game, the exact time and the titles of ongoing talks. At the same time, they picked up the signals of the r0ket and gathered information about the movement of participants at the convention. Privacy settings on the r0ket could be set to 0 (no radio broadcast), 1 (radio broadcast without identifying the user), 2 (radio broadcast including nickname). These information have been collated and made available to the public, so that various interfaces and visualisations of the database have been made. OpenBeacon and the physical design of the badges allows for determining face-to-face encounters with reasonable accuracy, which creates interesting possibilities for data mining. The most sensitive information is obviously who was talking to who at the summit, but statistics like how many people the average convention goer talked to can also be discerned. The problem of tracking participants using RFID have been raised by privacy advocates and watchdogs many times since the inception of the technology, most famously at the World Summit on Information Society organised by the International Telecommunications Union in behalf of the United Nations. After the first round (2003 Geneva, Switzerland) civil society participants officially protested the use of RFID chips in the conference badges and there was a promise from organisers to get rid of them for the second round.[124] For the second round (2005 Tunisia, Tunis), the promise was not honoured and participants were issues with RFID badges without knowledge. High-profile figures of the hacker

[124] The Washington Post cites Escudero-Pascual, Koch, and Danezis (2003) "The lack of security procedures violates the Swiss Federal Law on Data Protection of June 1992, the European Union Data Protection Directive, and United Nations' guidelines concerning computerised personal-data files adopted by the General Assembly in 1990, the researchers said." Hudson (2003)

world such as Richard Stallman[125] (2010), Bruce Perens[126] or Lawrence Lessig commented on the issue, while ITU flatly denied any tracking. It has been never turned out if there was a privacy policy in place, what data was collected, how long was it stored and most importantly who could access it. As in the case of the mass Pong experiment, the r0ket team rehearsed the tracking exercise with a crucial difference: the participants had enough information and prior expertise to give active consent to the tracking, policies were made clear and the data public. Through collecting the data publicly, they also demonstrated that others could have collected the same data covertly and use it for different ends. Finally, the hands-on attitude of hackers has to be notes: while privacy advocates noted theoretical problems, pointed out actual abuse and warned of hypothetical problems with RFID technology, hackers were happy to experiment with the technology themselves, acquire a working knowledge of it and find alternative uses.

The last contraption involving a r0ket that I found during the research period is called "Launching rockets with r0kets" performed at the µC3 on New Year's Eve 2011 (fpletz 2012). Hackers used the r0ket to short-circuit an electric plug, which emitted sparkles that kindled a match, launching fireworks to the sky.

We can conclude that r0kets were designed in a way that combined with each other or other components they could easily become parts of larger technological systems. In more general terms their functionality has been developed based on clues from the distinctive engineering culture of the social group it was destined to serve. *Technology did not figure in the equation as an instrumental solution for practical problems but as a form of sociality where the practice of development ties practitioners together.* However, just the fact that the exercise of technological creativity was framed by socio-cultural vectors did not mean that the artefacts developed with and through the r0ket did not work: indeed, the engineering aesthetics of hackers that values function as well as form lead to virtuoso performances of effective and efficient solutions by most mainstream standards of engineer in many cases. *It is only that these solutions have been more a side-effect of the social process than its end, therefore digital labour did not have to conform to the requirements imposed by capital accumulation or state control.* Indeed, while the hackerspace environment serves as a catalyst objectively through its infrastructure and subjectively through its social environment, it worked as a protected niche from the market and state pressure. The hackerspace, the hacker organisations and their conventions functioned as a hinterland for the relative autonomy of hackers that enabled them to navigate and negotiate with more powerful social actors on their own terms.

For Heidegger, good poetry is an act of gathering the truth and the good poet calls forth a people through creating a significant form. The r0ket team did just

[125] Hacker extraordinaire, developer of the emacs text editor, author and advocate of the GPL, the first Free Software licence.

[126] Hacker extraordinaire, developer of the BusyBox utilities, author and advocate of the Open Source Definition.

that by gathering traits of hacker cultures into the functionality and technical composition of a small scale electronic artefact that helped the proliferation of hardware hackers. Rather than an untimely intervention, the r0ket is an imprint of a life world. Yet it brought a new existential possibility to many hackers – the paradigm of physical computing. The operation questioned the instrumental use of technology what Heidegger calls enframing – a means to an end that at the same time bars the possibility of making truth appear to us. Consumer electronics are user friendly (Heidegger would say that they are ready-to-hand) but separate their users from the technology itself. The r0ket as an unfinished artefact opens the possibilities of technology to its users without being directly instrumental. We have emphasised before that this does not mean that an unfinished artefact could not work smoothly or function properly: indeed, the r0ket does a robust and reliable job at what it does: my r0kets are almost five year old now and saw much use, yet they are performing well. Yet the design opens up possibilities for exploration of technology itself, and through technology, the potentials of human experience – something that Heidegger calls poeisis, or revealing. We will return to this question when discussing the shadow of the r0ket two sections below.

9.7 Obsolescence

As the ambitions of companies like NXP Semiconductors – who wanted to promote its novel CPU architecture for a strategically significant user base – and Pollin – who wanted to add an ARM-based development board to its line of offerings even before hackers contacted them – show, the r0ket project was on time for becoming a massively popular product amongst hobbyists and their associated industries like industrial control, interactive media art, robotics, home automation, etc. In the eyes of the industry the potential of ARM processors clearly pointed beyond the capabilities of contemporary microcontrollers like the Arduino (based on the megaAVR ATmega line of processors) and the r0ket was an unlikely but still potential contender for its title as the go-to option for building small scale electronic artefacts. Moreover, even though the Arduino is as much OSHW as the r0ket, yet its designers lead a successful business based on its design, manufacturing and support (Banzi 2008). Finally, the differences in the production history of the first and second r0ket runs already pointed to commodification: perfecting the prototype based on “field test” results, the outsourcing of manufacturing and taking the product to the market. However, at least three factors blocked the r0ket from developing into a successful product.

Once, the r0ket is simply too crazy to sell well and widely: the rocket shaped PCB looks weird, the name tag is not suited for people with first, second or even third names, and despite the engineering appeal of a pure C platform, the vast majority of potential users are not familiar with that low level language.[127]

[127] Hackers say that for FLOSS hackers C is a low-level language because for instance it can access and arrange the memory directly, but for OSHW hackers C is a high-level language

Furthermore, the many extra features that the r0ket provides as a development board — the LEDs, sensors, display and wireless – are convenient for quick experimentation but at least one of those is superfluous for most actual use-cases.

Twice, in contrast to the Arduino developers who mostly worked on straight commercial projects, the r0ket team took on the project as part of their hobby. Their ambition was to develop their skills, serve the community and continue to do awesome engineering, not to develop business plans, compete on the market and make money. So one reasons why the r0ket did not become an actual product was that nobody wanted that. Of course, it would have been possible for another party to take on the challenge – for instance Pollin itself could have ordered re-runs using the OSHW bill of parts, schematics and documentation to negotiate directly with Etonnet. I have no empirical evidence to prove why things did not go this way but the other two reasons cited here must have been part of the equation.

Thrice, the rise of the Raspberry Pi generation of single board computers – mostly based on the ARM platform – superseded the need for microcontrollers altogether in the eyes of a large fraction of users. They became an instant favourite with geeks of all stripes simultaneously with their release in 2012 February, roughly six months after the first r0kets rolled out of the PCB factory. It turned out that the ARM architecture can provide the basis for complete computers, not just microcontroller boards. Many typical problems of physical computing could be solved now by throwing a proper computer with a fully fledged operating system in the mix. Of course for hard core hackers who value minimalism and economy of resources, such an approach amounts to nothing short of heresy, since it introduces a complexity utterly unnecessary. However, it gives users the possibility to attack the problem with their favourite tools – for instance web developers who already know how to manipulate websites with the JavaScript language can use it to read sensors and activate actuators. Since the price tag of a Raspberry Pi-class single board computer is not significantly different than a much more primitive device like the r0ket, the preference for the latter would have to be argued in largely aesthetic terms.

As a result, the r0ket largely fell out of use a few years after its inception. The conference programme browser application is still updated for major hacker meetings and the device is ubiquitous in hackerspaces where it sees some use for hardware hacking from time to time, but in effect it is a souvenir similar to a T-shirt that has more symbolic than practical value. Similarly, once the Chaos Computer Club helped to produce what is arguably the most technologically complex and fully functional electronic conference badge in and out of the hacker scene today, its members did not take up the challenge again. Therefore European hacker conventions does not customarily feature an electronic name

because it makes convenient to do the things that they normally do in architecture-specific machine code (assembler). A low-level language is reputedly harder to learn, harder to use and slower to write in because it makes explicit operations that high-level languages automatically hide.

tag.

9.8 Shadow: the r0ket versus the mobile phone

Since one of the principal questions is how the peer production of OSHW is different from mainstream consumer electronics, or in other words how hackers transforms technology, a comparative look at the r0ket and the mobile phone is useful to nail down the differences. While the r0ket has never been framed by practitioners as an alternative to the mobile phone, there are compelling grounds for comparison. Both are small scale electronic artefacts that one can hold in one's hand. Furthermore, they are equipped with sensors and input devices, computer-like capabilities for the logical manipulation of symbols, and of course use radio frequencies to integrate into larger scale infrastructures deployed territorially in order to communicate with each others and the Internet at large. Finally, the l0adables of the r0ket closely resemble the use of Apps (applications) on mobile phones for installing small programs that extend the functionality of the device through extra software and sometimes also hardware components in creative ways. Personal mobile computing is one expression to summarise such a technical constellation.

In terms of how the devices are framed, the infamous Nokia slogan of "Connecting people" resembles the arguments I have made about the r0ket, i.e. that its primary purpose is to bring together people. However, there are points of convergence too. On the one hand, the r0ket addresses a specific social group of people who actually meet each other and interested in the work of their peers rather than the liberal notion of people in general. This makes the ambition of the r0ket suitably smaller than Nokia's. On the other hand, the r0ket goes beyond the Nokia discourse by connecting people and machines together. The GPIO connectors are an especially articulate proof of including machines explicitly in the networks that the r0ket establishes. *Of course Nokia's practice includes many more and much more powerful machines in their operation, but the idea is to hide these from people through establishing a fluid user experience.*

In a certain sense the mobile phone is the arch-enemy of hackers. The most stunning aspect of the Levy (1984) account of the early days of hacking is the struggle of academics, hobbyists and entrepreneurs alike to realise the idea of the personal computer — a struggle that was mostly waged through sheer ingenuity and extreme persistence, but often verged on the illegal. The next generation of hackers fought for the freedom of software, enabling users to solve virtually any problem they encounter using FLOSS applications. Roughly around the time mobile phones started to proliferate it was already possible for the average middle class citizen to acquire a computer with reasonable processing power and run a free operating system on it. Even though the hardware is not OSHW, the IBM PC and most its derivatives are modular machines so at least incremental upgrades and repairs using spare parts are possible. Laptops are obviously less modular yet most hackers accepted that trade-off for the mobility gains. For many, the laptop looked like the end of a technological trajectory when weight and width were only determined by the size of the screen. Essentially, hackers were finally happy with a portable typewriter that could function as a remote terminal to more powerful machines, the high point probably marked by the release of the Lenovo Thinkpad X220 – the last one featuring the original IBM keyboards. At least two hackerspaces I followed (in Amsterdam and in Budapest) had members who made purchases of such machines with a look that future models represent the degradation of design. The new technological trend of netbooks, tablets and smartwatches were not fought for by the hacker community. Indeed, cult leader Cory Doctorow warned recently of the Coming War on General Computation (2012) where devices depart from the universal aspirations of the personal computer as the medium for the expression of technological creativity to specialised devices which are manufactured with a particular purpose in mind that they enforce on the user. Doctorow's keynote speech at the Chaos Communication Congress thus explicitly problematised what scripts are associated with devices and argued to keep the interpretative flexibility of devices as wide as possible. Meanwhile, mobile phones took over computers as the most popular platform on the planet for personal computation, receiving relatively little attention from hackers.

Mobile phones sport a stack of proprietary protocols so secretive that technicians reading them have to sign non-disclosure agreements. Despite the secrecy, protocols are flawed and conversations can be easily intercepted by authorities over the air. So-called "silent SMSs" which are invisible to the user can track their movements from afar. There are advanced announced every year towards a relatively open source mobile telephony solution, but the more independent research comes out it is more clear that mobile communications will never be as free as the Internet. Additionally, the Internet connections offered through data packages by mobile phone operators are are often crippled by a set of limitations. Characteristically, the sheer speed of the connection is not a priority for hackers. On the other hand, they are put off by ideas such as banning the peer-to-peer BitTorrent file transfer protocol from mobile platforms, inserting proxies that optimise the bandwidth through changing the packets over the network, or imposing a cap on the amount of data that can be downloaded.

The product design of mobile phones do not offer many excitements to its users either. The guiding light of contemporary consumer electronics is the Human-Computer Interaction paradigm loosely inspired by Heidegger's ready-to-hand concept: tools should be invisible for their users, who should merely focus on what they actually want to do rather than how to do it. Thus ends win over means and what we get is nondescript beige boxes or sleekly polished chromatic mirrors separating us from the supposedly amazing machines. Packaging prevents users from tampering with the machine and protects against ambient hazards like weather or accidents. The minimalism of presentation guides the users along scripts that lead them beyond the device with as little friction as possible. The minimalism of technical composition in the r0ket includes the design decision not to put a ready-made case on the device. The exposed PCB calls attention to the technological artefact itself, with all components receiving the same exposure. There is no particular accent to draw the attention of users towards the display, the joystick, or other input/output devices: a sensor or a memory chip is considered just as important. It is trivial to see that the r0ket is not designed as a black box, but not even with the more subtle interactive visual rhetorics of consumer electronics. For the kind of users suggested by the design decisions behind the r0ket, each functional component in the machine could be a starting point for interaction with it. Maybe it is necessary to fix a transistor that broke off? Maybe it is interesting to read the contents of the memory through wires connected directly to its tiny metal legs? Maybe writing a new l0adable requires the programmer to know the exact type of light sensor on the device? Any of these interventions are invited by exposing all components at once and letting the user get to know them. After all, most users have to actually grasp the basic technical composition of the device before they can do as much as finding the joystick to interact with the r0ket's screen. While that is definitely a flow from the point of view of ruling paradigms in Human-Computer Interaction, it is also an opportunity to bring users closer to the technology itself: to explain what is it that they hold in their hand and how it works.

In terms of hardware parts, mobile phones are less modular than even computers and much less popular than r0kets. As described in the hardware section above, the r0ket team tried to use off-the-shelf components or ones that can be easily found and bought by end users on the market. This is significant because most mobile parts are not accessible on the market for individual users who want to acquire spare parts. Moreover, the fact that r0ket designers used some depreciated mobile phone components – like the Nokia 1110 LCD display – shows that mobile phone components have been more standardised in the past. In the beginning of the millennium it has been a common complaint of users that each subsequent generation of mobile phone came with its own charger, and of course mobile phone chargers between brands have been incompatible with each other. The European Commission stepped in to intervene in market dynamics through the design and advocacy of the common electrical power supply standard (European Commission 2010).[128] The reasons EC cites are twofold. On the one hand, users have to buy a new charger for each time they get a mobile phone, which is highly inconvenient. In the other hand, old chargers are not useful any more and become toxic waste. While not a binding legislation, the campaign by the EC was more or less successful in tackling the problem. However, it is telling about the contrast between the technological orientation of the mobile phone industry and the r0ket developers that the regulator had to step in to do the right thing that is obviously better for users. As shown above, the r0ket

[128] The actual standards referred to here are the European Committee for Electrotechnical Standardization EN 62684:2010 and the closely related IEC 62684:2011 from the International Electrotechnical Commission.

designers took care to maintain compatibility with rival products of the same class like the popular Arduino development boards.

The r0ket itself cannot connect to the Internet at all, but at least its networking protocol (OpenBeacon) is simple, open and extensible: the very qualities that make Internet protocols interesting for hackers and the lack of which makes GSM (mobile) communication practically impenetrable. Again, OpenBeacon and the conference setup in which it is often used allows fine grained tracking of hackers, but at the same time it allows a set of privacy options to avoid it too, and these options are implemented in the r0ket user interface. The openness of the software, hardware and protocol means that interested parties can verify if the privacy settings advertised are actually working correctly.

Mobile phones are an integral part of large technological networks built and maintained single-handedly by mobile phone operators: they are designed to function in dependence to cell towers which have to be deployed territorially. While hackers do deploy similar specialised antennas for tracking r0kets at the most important conventions, these are not tightly coupled with the devices which means that r0kets function perfectly in the absence of antennas. r0kets talk to each other directly through their meagre antennas so they are completely autonomous on the field. Therefore, they do not depend on a centralised organisation with control over a territory that has to deploy costly background infrastructures to enable communications.

It is obvious that the capabilities of the r0ket are by and large inferior to those of mainstream mobile phones that came out in the same year. This is not a surprise because mobile phones are designed to serve whole societies while r0kets were made to entertain a couple of thousands hackers. Nonetheless, we can conclude that hackers are happy to put severe limitations on a technologies they embrace if at the same time they can make them simple, standard and extensible. As long as the design is kept open in a variety of ways that I tried to capture with the unfinished artefacts concept, the limitations of the initial design are considered an advantage, because it can involve more people and possibilities in future developments. Moreover, as the practices of the demoscene demonstrate, limited hardware architectures can be developed later on by pushing their limits through clever usage and modifications. In these cases the expertise of the users can broaden the interpretative flexibility of the device without having to change the "hard" parts of the hardware.

Finally, the usage patterns of the r0ket can be assessed through looking at how it functioned embedded in the fabric of everyday life in the particular social context of the hacker conventions. The LCD backlight and the flame which could work as a minimal torch are actually useful in the course of these 24/7 events. Participants can easily see each other's nicks even during parties and other events that happen throughout the night, and in Camp it essential to have a torch for finding one's tent or toilet in the camping area. Announcements by various groups used the r0ket to spread calls for their events, a fraction of participants used it to exchange contact cards and cryptographic keys with each other, as well as sent personal messages, and many played collective games put together similarly to the Pong presented above. According to my observations, the fahrplan (conference programme) was the most widely used default l0adable at conventions. To bring these together, it can be stated that during hacker conventions the r0ket complemented many major functions of mobile phones: messaging other people, playing simple games, getting information. These interactions were confined to the territory of the event – but then again, even other ICTs like mobile phones are most often used to reach out to people one meets regularly in person in one's life world.

Therefore, the r0ket could be seen as a reminder that the technological trajectory of the mobile phone is not the only possible and necessary technological trajectory that personal computing could take. Moreover, in contrast to other alternative proposals, hackers actually implemented their own small scale electronic artefact to counter hegemonic concepts of modern technology, rather than merely coming up with an alternative vision as a scholarly critique, a new media artwork or a manifesto. Comparing the resources that the mobile phone industry can throw at problems and the resources hackers could mobilise through their peer production model, it is altogether impressive that an actually working artefact could be made in reasonable numbers to equip the scene. The r0ket gave exactly what most attendants expected to find at a hacker convention: a technological experience that brings humans and machines closer together. *The peer production model coupled with the relative autonomy of the hacker scene allowed the r0ket team to realise their plans without trying to be competitive on the market or fit into the discursive frame of state subsidies.*

9.9 Conclusions of the first case study

Castells calls similar devices as a mass self-communication device (2009, 116–136; 2012, 230; 2014). Mass self-communication is a paradoxical expression that seeks to conceptualise an inherent contradiction of late capitalism. As many other contradictions of capitalism, it is based on a gesture of separation followed by a gesture of unification. The aspects which are lost between these processes of separation and unification often constitute the starting point for critiques of capitalism. The classic example is the widespread alienation that follows from separating of the worker from the means of production on the one hand, and the product itself on the other hand. The capitalist owns the fixed capital required for work and buys the human time of the worker to make products. The work of unification is carried out by market actors that allow the workers to buy products made by themselves or other workers, using the money they got for selling their work time. In late capitalism there are specific processes of alienation that follow from the principles of liberal democracy. These are described by Giddens (1991) and Beck (1992). In broad terms, these force people to act like individuals, think about themselves as individuals, and express themselves as individuals. The counterpart of these individuation processes is the networking effects of ICT usage patterns. Using ICTs, capital unites these individuals in instantly mass markets through standardised interfaces that greatly simplify interactions. The constant production of the self through these devices re-establishes social ties according to the requirements of capital accumulation.

In order to counter technological determinist discourses these changes should be attributed not to the technology itself, but the human desires which spawn these technologies and the usage patterns people find for them. There has been no "evolution" of technology where one generation of technologies logically followed the next one and the market ensured the survival of the best models. Rather, technology changed in tandem with changing social relations, responding to social conflicts where people struggled with the contradictions of capitalism.

The technical composition and usage patterns of the r0ket can be understood in the context of these transformations, and they actually follow them in broad strokes. However, some of the key differences between mobile phones and r0kets also point to local divergences which expose social contradictions as they are inscribed in commodity electronics. Therefore the divergences between the particular technological repertoire of hackers and the mainstream technologies can be read – much like cultural studies read popular material culture – as the material residue of critique coming from a particular social group. The ideas and practices that characterise the construction and use of the r0ket show points of contention where practitioners with enough expertise to shape technologies but also enough autonomy to make critical assessments challenge what mass self-communication devices should be and how they should develop. They draw alternative trajectories for these technologies which open up possibilities for interventions.

r0kets are individual devices in the sense that individuals get, buy and use them, and their most important function is exactly self-communication: to display the name of the person to others facing her. Moreover, the name is a short nickname that is more individual to the individual than the names on their state-sanctioned personal identification documents. While family names are determined by lineage and first names by parents, nicknames are chosen by the person or bestowed on them by the community, answering to a logic that finds the essence of a person in their biography rather than their predecessors. The r0ket allows for configuring the font and optional animation that is used to display the nickname – another feature that caters for self-communication. In this sense the small screen of the r0ket is not unlike a social media profile which allows for the production of subjectivity. However, the r0ket goes head over heals to push self-communication possibilities further by a proliferation of functionalities that invite the user to modify the socio-technical conditions under which it happens. Moreover, as a counter-tendency the r0ket allows for hiding identities through privacy settings, setting false names, and finally putting down the device to built it into an impersonal artefact – for instance a remote control car comprised of three discrete r0ket. It is interesting to remember that at this point in history in most developed countries it is non-trivial to get an anonymous SIM card and organisations from states through banks to social media websites are imposing real-name policies along with identifying users by their phone numbers. Under such socio-technical conditions the fluidity of the conditions of self-communication in the r0ket is not merely the rehearsal of hegemonic patterns but also opens up possibilities for questioning them. In conclusion, as a mass self-communication device the r0ket supports a wide range of self-communication but rather than providing an rigidly scripted technological framework for doing so, it targets the communication technology itself as a self-expression, so that users can either bend it to extend or to restrict the possibilities of self-communication.

On the other hand, in contrast to mass media, r0kets do not directly address a deterritorialised mass audience: it is rather a local broadcast tied to specific events and communities, times and places. In actual fact most of what r0kets can communicate reaches as far as eye contact and even the wireless does not carry further than a clear voice. Therefore, it may well be that the r0ket rather tightens communication loops between bodies rather than abstracts them away as a mass communication device. In such capacity it augments presence by multiplying connections around points close to each other rather then dispersing them. Furthermore, as the blueprint of an engineering subculture r0kets highly overcoded with collective imagery and engineering solutions that reflect a shared aesthetic code – arguably leaving less space for the communication of individual selves.

On the level of alienation in the production process, the biography of the r0ket shows the desire of hackers to own the fixed capital that allows for production, as well as to create commons which can function as distribution channels beyond the market. However, it also shows how these ideals cannot be realised in a pure form in the case of OSHW – that is why the r0ket team have to exercise their relative autonomy for negotiations with more powerful market actors. The result is that in reality most of the human labour that went into the r0ket device was reminiscent of classic factory production, yet the actual work inside the r0ket team was not coordinated through a capitalist management model, but lead by a few core people who enrolled others to the project in various capacities in order to collaborate towards a common goal. Ironically, in this case the phase of production that interviewees highlighted the most was the sweatshop-like mass assembly sessions which they perceived as a heroic undertaking that should and could not be repeated a second time. Indeed, if models proposed for the production of OSHW are pushed too far, they often hark back to pre-factory piece-work rather than the medieval craftsmanship that is emphasised by theoretical accounts like Sennett (2009) or Raymond (2000) (in the latter see the chapter entitled The Joy of Coding). Adherents of peer production would argue that problems of scale such as that presented to the r0ket team are simple to address by distributing the production throughout a large territory of local production sites. Since there is no empirical evidence on record for such a model of distributed mass production in operation for OSHW today, I do not dispute that argument. However, in the second case study I do present a case in which production of OSHW is massively distributed geographically – and show how mass production is not viable under such circumstances.

In the r0ket's case the "pilot project" of mass production went beyond the infrastructure facilitated by the local hackerspace. µC3 is very well equipped for prototyping, experimentation and crafts-like production of small scale electronic artefacts — but it is not a factory. When mass production of identical copies is needed, there is also a need for the factory. Peer production is possible yet unlikely in the factory, and necessitates the workers to be in possession of the fixed capital. Cooperatives and occupied factories point in this direction but are not the subject of this study. In this study, peer producers could enroll a capitalist in the project, and therefore acquired control of a factory (fixed capital) for as long as they need. We can note that this is similar to the revolutionary custom of taking over printing shops and presses temporarily for manufacturing propaganda.

What does these limitations and possibilities mean for the peer production of OSHW as opposed to FLOSS? Hardware is definitely hard. Production of hardware is obviously more capital intensive and especially requires a bigger initial investment. Its production requires closer engagement with the organs of the capital and to a lesser extent the state[129]. Successful mass production of OSHW is only possible at a larger scale than the hackerspace. While FLOSS production can be a solitary activity that does not involve any monetary transactions, and therefore takes place in its entirety outside of the market, OSHW production requires substantial amount of money and engagement with the market. Since most peer producers do not have either the capital to invest or the fixed capital to mass produce, they have to look for ways to involve market actors in their operation that point beyond the default social relation supported by the market: monetary transactions. The r0ket hackers were able to enroll market actors in the project in a way that enlarged the definition of the market to account for externalities like ethical relations, community management, and so on. They therefore helped in the emergence of cognitive capitalism which thrives on externalities but also subverted the old order of the market where actors are simply profit-maximising rational and individual actors.

In summary, what we have learned about peer production of OSHW in the hackerspaces? First, commons based peer production of OSHW has not been completely possible despite the motivation of the actors because hardware production required seed funding and fixed capital to produce. Second, OSHW is rarely open source because crucial components are often proprietary – what is open source about OSHW is mostly the documentation of the composition of

[129] I choose not to expand on engagement with the state but designers had to conform to a host of industrial regulations – so much so that one of these would possibly make the r0ket as a mass commodity inviable.

components, not the components themselves. Finally, as long as it goes above a few dozen copies, it seems that the peer production of OSHW meets the infrastructural limitations of the hackerspaces. Of course, the infrastructural limitations of the hackerspaces are only a reflection of their political economy: how much capital can members mobilise, how much participation to motivate and as the derivative of the two, how much they can extend their relative autonomy. Therefore, the challenges of OSHW production can be a catalyst for organising on a higher level of aggregation across, between and beyond the hackerspaces. The role of the Chaos Computer Club in the production of the r0ket is an excellent example.

We have also observed the marginal role of licencing in the identity construction of OSHW practitioners which stands in stark contrast with FLOSS. In the case of the r0ket licencing the hardware was more of an afterthought: the open licence added to a vague place inside design file in a proprietary file format way after the completion of the project. The role of the principles of composition is much more central to the process, where we observed simplicity, modularity, standardisation and specific functional features which keep the functionality of the artefact open, thus thwarting closure and stabilisation. Beyond the actual artefact the second most important factor was the quality of social relations that the designers foster For instance the r0ket would not have become an eminent example of OSHW if not because of the tireless support of the r0ket team for their users. Here is the key: first and foremost, the r0ket is *not* OSHW because of appropriate licencing but because it is immersed in a set of social relations that not only allow it be changed but actively support changing it. In fact the same people would most probably support changing the artefact even if the licence would explicitly prohibit it – and call it reverse-engineering.

The only problem with such an understanding of OSHW is that it goes beyond the actual OSHW definitions out there. That is why I proposed the concept of the unfinished artefact which can capture the understanding of OSHW as a set of social practices – which may or may not be protected by legal instruments. After all, the whole point of the legal instrument has always been to promote a particular set of social relations around an artefact. In writing the biography of the r0ket badge I came to the conclusion that peer production in the hackerspaces is a particular kind of sociality where the actual artefact is merely its material residue. Such formulation explains the relations between unalienated labour (i.e. production of small scale electronic artefacts) and the purely discursive interactions that will up so much time in the hackerspaces. Furthermore, it also accounts for the complaint by makers about hackers that "hackers never finish anything". If material production is merely the residue of the production of social relations, then it is easy to see why hackers would happily abandon a project or move onto another one without turning prototypes into products. In fact, the practice of the peer production of OSHW only makes sense as a critique of alienation as long as it is not the merely a foreplay of free labour for the product development of commodities.

While the principles of peer production have been inscribed in the artefact during its design process mainly through the selection of parts and the documentation, references to hacker culture were worked into it through providing the content and context of applications. Throughout the biography I emphasised that both the process and the product brought different kinds of hackers together with each other and towards working on OSHW. In terms of the democratisation of technology it is striking how a social group could develop an artefact – and a particular understanding of technology – which incorporates its values on so many levels while straying away from mainstream strands in so many ways. However, these impressive results should be seen in a more nuanced context: it is not just any social group who managed such undertaking but an engineering subculture comprised mostly of exceptionally privileged people. Social groups which put forward alternative practices of technological creativity or production are typically marginal or marginalised, characterised by some special social, cultural, geographical, psychological or physical deficiency: (ex-)colonies, the global poor, slum dwellers, the elderly or the blind. Hackers are hard to fit into such a scheme. Despite their diversity, these self-designated experts can be more or less described demographically as able bodied young or middle aged urban white male middle class technology professionals. As a result, even though the concept of technology they put forward may be puzzling for most other people, it is not automatically delegitimised, which leaves open an interesting space for critique.[130] An alternative conception of technology can be potentially stronger if it is put forward by an eminently privileged social group.

Haché (2014) described the idea of technological sovereignty which resonates with the practice of a social group producing its own technology. In her formulation, technological sovereignty is the power of a community to decide what kind of technology it needs, how that should be produced and to acquire such appropriate technology. Notwithstanding its obvious limitations, the r0ket badge is an example of hackers inventing, making and using their own devices. However, it is clear that hackers are in a special situation because it is a social group almost exclusively comprised of people with engineering expertise, therefore it is easier for them to develop or at least gain control of their technology than for other (less privileged or less specialised) social groups. Haché (2014) derives the idea of technological sovereignty from the concept of food sovereignty as it has been proposed by the Via Campesina social movement. Even though food sovereignty is as important as technological sovereignty, hackers would have a much harder time developing their food sovereignty. They would have to work together with other social groups and communities to develop different kinds of sovereignties together.

The example of Club Mate is telling: in a certain sense they arguably developed

[130] Even though hackers often encounter challenges of delegitimisation in the form of criminalisation and smear campaigns which cast them in the role of antisocial basement-dwelling freaks, evil mad scientists, or both. However, these characterisations do not necessarily question their technological expertise per se, but casts shadows on their role they play as self-designated experts.

their drink sovereignty – without hackerspaces their favourite drink would only be sold in its native Germany, but now there are distributors as far as Canada and the US. The hackerspace in Budapest, Hungary (H.A.C.K) started to serve Club Mate and it quickly found a local distributor which now sells to several dozen locations across the city. As in the r0ket biography, the development of sovereignty actually worked through a transversal intervention into the market. Hackers as a social group decided that the culturally appropriate drink for them is Club Mate, and they have been able to make sure it is distributed in their territory. Many hackerspaces also started to produce their own versions of Club Mate too — at the Hack42 hackerspaces in Arnhem, the Netherlands, there are 42 varieties available at the bar. However, in contrast with technological sovereignty where hackers arguably challenge hegemonic assumptions about what technology should be, how it should be made and what it should do, in the case of Club Mate the thrust of the gesture is barely different from any subculture where members assert their cultural identities through consumer choices that effectively diversify the market by sustaining another niche segment.

For the same reason of their technological expertise, Feenberg's subversive rationalisation thesis puts hackers in a peculiar spot too. The author posits subversive rationalisation as a challenge to the technological determinist ideas and practices of "dominant social groups at the level of design and engineering" by the "subordinated standpoints of those dominated within technological systems" (Feenberg 1992). While the hacker tradition is an engineering subculture that have been instrumental in some of the essential moments of computing history, it is not a dominant but a rather subordinated stream. Hackers routinely complain about the bad direction that industry practices took and intervene in various ways to set it right. They do face repression from the state and recuperation by capital, but it is still a stretch to designate them as being "dominated within technological systems."

Yet the most important characteristic of subversive rationalisation for Feenberg is that it questions the unilateral course of technological progress as well as the imperative for social institutions to adapt to "the technological base" (Ibid.). From the second perspective the r0ket is an eminent example, especially in contrast to mobile phones which impose themselves as the only choice in personal mobile computing. The hackers' invention re-imagines personal mobile computing in a totally different direction, guided by an alternative concept of technology based on open standards, off-the-shelf parts, simple design, peer-to-peer networking and territorialised functioning.

What we can learn about the political potential of subversive rationalisation from the object biography then is that especially when it comes to hardware (small scale electronic artefacts), it takes a privileged social group with a relative autonomy to exercise subversive rationalisation beyond coming up with an alternative vision of technology or alternative usage patterns. In fact manufacturing requires enrolling some parts of the industry in the project even on the modest scale of the r0ket badge. I tried to show in section 7.7 how the hacker scene have built such relative autonomy through capital accumulation, institutionalisation and cultural production that can serve as the material and social basis for subversive rationalisation.

In my view the ambiguous position of hackers vis-a-vis technological sovereignty or subversive rationalisation goes back to if and how we conceptualise them as experts. The r0ket story shows a kernel of experts doing the core of the development, yet the thrust of their work is exactly to lower the barriers between experts and non-experts. More precisely, they are working to make technologies that require a lower level of expertise without giving up the idea that users should be able to understand, control and modify the tools at hand. Participation in the project is attracted by expertise, at which point the making of the r0ket diversifies from a research project to include a strong component of education too. In fact without blending research and education the manufacturing aspect could not succeed because the work force would be lacking. Moreover, in the terminology of Collins and Evans (2007) the expertise of the kernel of the r0ket team is recognised by their peers not through the meta-criteria of credentials but through their track record, that is their past contributions which worked. In terms of peer production, we can say that the process that leads towards the common goal of production can easily accommodate or rather strictly require research and educational elements, especially because participants typically lack the necessary knowledge and skills for the job, as well as the necessary resources to implement their ideas without doing at least a bit of research.

Considered from a long view of the trajectory that technologies trace it can also be argued that the r0ket is not merely an alternative pathway of personal mobile computing straying away from the mainstream idea of mobile phones but it is actually enforcing the path dependency that was there in the modular IBM PC compatible desktop computers on a newer wave of technology. In other words, the r0ket could be seen as a "computerised" mobile phone. This is counter-intuitive because from this perspective the hackers' reconceptualisation of technology is retrospective rather than avant-garde, conservative rather than subversive. Nostalgia for the achievements of the Ur-OSHW movements in the construction of the first personal computers and the FLOSS movement in the construction of usable, understandable and modifiable operating systems becomes a political force technologies which hackers see in many ways as decreasing the agency of users. As a result, the technological frames produced by the r0ket team for personal mobile computing emphasised some of the same qualities that were valued in early personal computers: they view their technological artefact as an educational and research tool, as well as a vehicle for recreation. I showed how these values have been encoded into the functional composition of the device itself through applying design principles and selecting particular parts.

The results of the r0ket analysis qualify Shirky's thesis that peer production practices can be characterised as *mass amateurisation.* There are two mutually reinforcing and mutually recursive processes to be observed in the case of OSHW where civil society and market actors establish a hobbyist ecosystem. The role of civil society is mainly in the democratisation of expertise in the area of small scale electronics, which is realised through the medium of hackerspaces as the infrastructures of peer production. While hackerspaces serve as the sites for the cultivation of expertise, their open door policies encourage participation in the co-creation of such expertise – and as I have previously argued, small scale electronic artefacts are but a residue of such a sociality. As von Hippel (2005a) argues, in the first phase the very possibility of hobbyists taking up microcontroller programming is enabled by changing market conditions that allow them access to a range of basic components through the open market – components that were previously only marketed between manufacturers.

Increased participation in the cultivation of such expertise in turn creates niche markets of its own, from individual hackerspaces selling kits at their workshops like the TechInc logo described in the forthcoming Section 10.1.5, through small companies offering PCB manufacturing and component placing facilities for their local region like Leiton or E.E.P.D., to companies supporting the whole OSHW manufacturing process in the context of global capitalism such as Etonnet. These offer individuals and groups an opportunity to learn the basics locally, try out their ideas with the support of local industry, and implement mass production in a global framework. While the knowledge component proliferates in a knowledge commons according to the principles of commons based peer production with the participation of both grassroots entities like hackerspaces and market actors such as component vendors, when it comes to tangible components the production process will pass through the market, starting with major microchip manufacturers and often going through SMEs operating regionally or globally. Therefore the democratisation of expertise creates new markets. If a hobbyist scene in symbiosis with such a market can be called *mass amateurisation* is dubious for two reasons. On the one hand, the production of small scale electronic artefats still requires a specific skill set. Even though that skill set can now be taught to children and adults without a broader engineering background, given the material infrastructures to facilitate such teaching, these amateurs cannot be mentioned on the same page as Shirky's Flickr users who push a button on their smartphones with the icon of a mass media monopoly on it (2008). Participating in hardware hacking do not require taking courses and exams and getting official certifications like amateur radio operators, but it does necessitate familiarising oneself with an alternative engineering culture including its techniques and social norms. Therefore, there is a democratisation of expertise but calling it mass amateurisation would be saying too much. On the other hand, in Shirky's case the argument is that Flickr users are encroaching on the market of professional photographers and agencies, while in the case of the r0ket we see peer production practices around OSHW actually expanding the market. Therefore talking about the mass amateurisation of the hardware industry would be once again going too far, it is clear that access to fixed capital through the market and through cooperative structures like the hackerspaces lowered the barrier for participation. To put it more bluntly: appending a niche market to the hardware industry is not a *mass effect*, while teaching people to solder is education and not *amateurisation*. Therefore, instead of mass amateurisation, in the case of OSHW it is better to write about the democratisation of expertise and lowering the barrier of participation in production.

In terms of the changing patterns in group formation it can be argued that hackerspaces allowed for *group formations* for the cultivation of particular socialities which are characterised by technological productivity – once again, as a side effect. However, while Shirky argues for the novel character of these "architectures of participation" (Humble, Molesky, and O'Reilly 2015) which drive down collaboration costs and thus free market logics to operate, I would rather emphasise the role of informality in these collaborations and the continuity between technological production and other forms of social interaction. I can do this because in contrast to Shirky's case material of online platforms which put a great amount of hardware and software between people, mostly so that they do not have to meet each other to collaborate, hackerspaces as material and communication infrastructures of peer production aim to actually bring people together so they can collaborate more effectively: that is without technical mediation. In the same manner, legal organisations are set up to interface the social logic of the hacker scene with larger social structures such as regimes of accumulation based on property ownership (e.g. landlords), but the main thrust of these official organisations is not to introduce hierarchical structures into the communities or to enforce work discipline – it is to provide a shelter for members from repression and market pressure. It is tempting to argue that productivity in the case of the r0ket was not the result of an inherent efficiency in the socio-technical architecture of peer production (falling transaction costs) but questioning the institutional and contractual bonds and incentives that characterise the modern division of labour. Indeed, it would be too much to suggest that r0ket production in itself was efficient in any conventional sense of the word. It did not work because of a superior economy of resource allocation: it worked because the r0ket team found many ways to mobilise resources as needed. While certainly too metaphysical, Bataille's notion of an infinite source of energy that can be tapped by life itself comes to mind. Since the team could frame the r0ket as a community project, it could enroll a diversity of actors in the project from a corporation which donated free components, through a factory that donated the use of its fixed capital, to volunteers who did the manual labour. None of these have been possible if r0ket production have been part of a product development project in a commercial setting – and would have been less attractive if they try to pull it off as an educational initiative.

As for Benkler and others claims about peer production being another way to organise the economy, or even a new industrial revolution (Anderson 2014), the case study of the r0ket underlines the critique mounted by Dickel, Ferdinand, and Petschow (2014) who argue that shared machine workshop facilitating peer production are but a protected niche for experimentation in innovation networks. The r0ket could not have been produced without conventional factories and a global division of labour between core and peripheral countries organised through a capitalist market. Even if we extrapolate the results to gauge future prospects, it is hard to see how the hacker scene could have replaced these facilities with their own resources. In this sense Benkler's notion of commons based peer production as a practice exclusively tied to the production of intangible goods held: only through negotiating with the external agents could the r0ket team produce a working piece of hardware at some scale. Benkler argues that whenever capital accumulation comes into the picture, the purity of commons based peer production gives way to hybrid implementations that combine peer production with business models that are orthogonal to it. At the same time it is notable how the relative autonomy of the hacker scene, as well as peer production practices allowed them to negotiate the conditions for the hybridisation of the process on their own terms.

I noted earlier how Bauwens' interpretation of peer production shifted from an autonomous interpretation that conceptualised peer production as a social practice that cannot be captured by the division of society into capital, state and civil society to a post-autonomous interpretation where peer production was packaged to update or at least complement the organisational regimes of capital, state and civil society – an update legitimise through technological deterministic arguments posing their transformations as a historical imperative. In fact the case study of the r0ket badge shows a similar trajectory of peer production practices cultivated in the hackerspaces scene overflowing to transform the relations of citizens, consumers and activists to the state, capital and civil society. The main significance of the r0ket have been the promotion of a different relation to technology through its manufacturing and usage. Such concept of technology did not emerge spontaneously through the unilinear development of technology as technological determinists would have it, but it was actively produced and cultivated by the hacker scene *against* mainstream engineering practices. This does not in itself make it a progressive social force especially because many of its characteristics follow the unfolding logic of cognitive capitalism – for instance capitalising on externalities that have not hitherto been considered significant factors in economic and industrial relations. However it *is* an instance of a social group rejecting the technological imperative and coming up with their own direction where they want to develop technology and their relation to electronic artefacts. For hackers, engineering practice is about creating technology that shapes society, not adopting society to the unilinear development of technology. While all technologies can be seen as an intervention in social relations, other engineers often understand their role as impartial, effectively implementing goals which are articulated by other social groups.

While the biography of the r0ket badge included a single hackerspace as the main site of organisation and production, and usage mainly took place at hacker conventions, the second case study explores a distributed practice of making small scale electronic artefacts in many North European hackerspaces. In contrast to the r0ket badge, the door systems explored in the next chapter are usually not licenced explicitly at all, so that they show how the social practices around OSHW can function effectively without legal instruments or definitions in the confines of the hackerspaces scene. These observations can flesh out the concept of unfinished artefacts which was posited in this chapter. Moreover, having established the idea that some hackers cultivate an alternative conception of technology that stands apart from technological deterministic models, and having shown how it is encoded in small scale electronic artefacts, the case study of the door systems allow for the exploration of the opposite effects: how the values encoded in artefacts shape social relations in the hackerspaces. These results can contribute to the development of the concept of unfinished architectures: infrastructures that facilitate peer production. Finally, the door systems as hackable access control systems are an excellent site for the investigation of how expertise and participation are co-articulated in the hackerspaces milieu.

10 Open hardware case study II: Door systems

A door system is most often a button near the door that is used to indicate when the hackerspace is open. In a typical scenario, the first person who arrives pushes the button, and as a result an image appears on the website announcing that the hackerspace is open. The last person to leave should push the button again, changing the image on the website to say the space is closed. Such system exists are installed in North European hackerspaces included in my research (see section 4.4) and in many others. The actual systems always go beyond the basic functionality described here. As a rule, while the hackerspace gets older its members develop the door system in the most arcane ways, and door systems start to merge into large technological infrastructures.

It is not an emic expression: in fact there is no canonical name for door systems even if they are a widely recognised and replicated practice across hackerspaces. Furthermore, there is no master blueprint for making a door system, or genealogy of door systems that identifies their inventors or origins, as in the case of classic OSHW like 3D printers, for instance (see Figure 13). Door systems exhibit family resemblance because partial ideas – but not full implementations – are imitated from one hackerspace to the other.

I started to study door systems for three main reasons. Once, exactly because it was simply something people did in hackerspaces without thinking much about it – it blended into the landscape. Twice, because it was so unique to the hackerspaces scene that I have not seen it anywhere else, and it was a type of artefact that is deeply tied up with the organisation of the club, so I thought that was a good way to capture the particularity of the hackerspaces milieu. Thrice, because being a distributed practice I felt that it is a strategic case for establishing the fact that eminent OSHW – that is an unfinished artefact – does not have to be tied to licences, rather to the social relations that sustain it.

10.1 Diversity of implementations

Presenting a panorama of door system implementations in this section, I would like to establish three points. First, the simple fact that hackerspaces across Europe have door systems whose basic functionality is highly consistent, thus they can be taken as a valid unit of analysis. Second, despite highly consistent basic functionality, technical implementations of door systems shows great variation across hackerspaces. Third, that technical variation can be accounted for in terms of the specific cultural differences of hackerspace memberships which could be linked to differing aspects of hacking, as much as the geographically specific material conditions from the physical properties and location of the hackerspace building to the political-economy of the host city. I put forward the first claim as indisputable ethnographic data that serves as the basis of the analysis, the second claim as a well-grounded observation from which multiple more theoretical

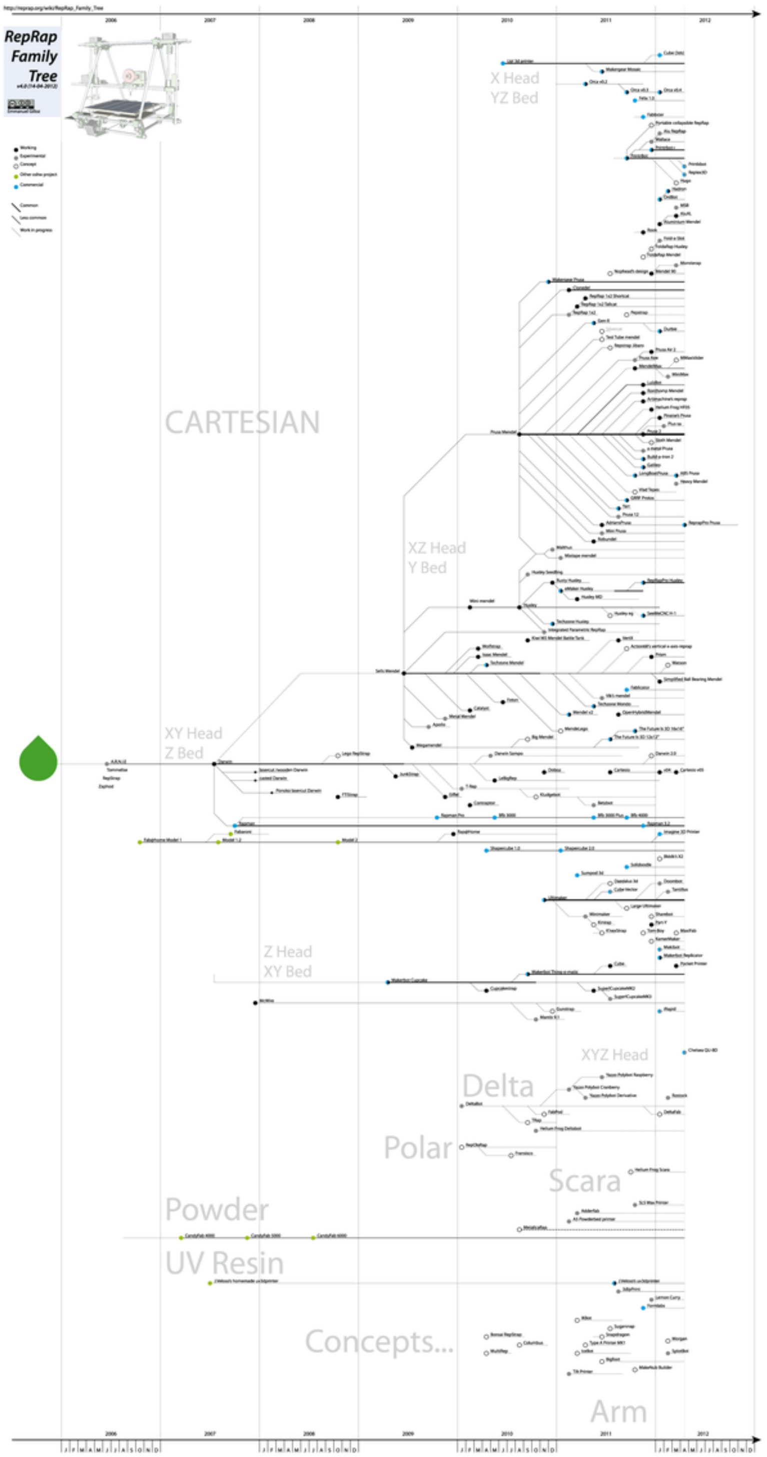


Figure 14: RepRap Family Tree, 2006-2012. Author: Emmanuel. Licence: GNU FDL 1.2. Source: http://reprap.org/wiki/File:RFT_timeline2006-2012.png

arguments will follow, and the third claim as a tentative hypothesis which may or may not hold strongly in particular cases.

10.1.1 Bitlair, Amersfoort, The Netherlands

Bitlair is the home of the most immaterial door system I encountered, even though as we will see further on, the work on door systems here had a strong influence on the development of door systems into unfinished architectures and large technological systems. It merely relies on identifying known users authenticated with the wireless access point, although it does contain a custom PCB, electronics and wiring between the access point and the in-house server which does most of the logic.[131] However, even the association check is performed against the Space Federation shared authentication system, which enables wifi users from one hackerspace to seamlessly authenticate with access points in other hackerspaces and hacker events (see section 10.4.2). It was conceived by Bitlair members collaborating with people from other hackerspaces, just as the SpaceAPI which makes all the different door systems in the Netherlands – and to a lesser extent, around the world – compatible with each other (see section 10.4.3). While the Federation is a younger project, the SpaceAPI is already an indispensable part of the networking practices of hackerspaces. All the door systems touched upon here make their outputs available through this standard, which is striking given their totally different implementations.

From all the hackerspaces I visited, it is only in Bitlair where the original social function of door systems is key to the operation of the hackerspace. Normally all members are entitled to a key and they are issued one, but at Amersfoort only a few key-masters possess the physical keys to the building, so other members have to check and arrange with them the opening and closing of the space. The fact that this door system does not have a physical interface such as a button or a switch installed on the wall near the entrance prompted three observations.

First, that it is very important for most spaces to manifest their technology in a highly visible, haptic and straightforward way. On the one hand, I understand that this is in line with the hackers' conception of technology which should lend itself to investigation and be clear about its operation. On the other hand, supporting the hypothesis posited in Chapter 7., hackerspaces are specialised in hardware hacking, and placing the button prominently near the door sends a message to anybody who enters the room that there are hardware hackers around who can build their own idiosyncratic contraptions.

Second, a recurring explanation I received from members of other hackerspaces about the necessity for an actual button instead of automatic sensors was that users have to be in control of the technology, especially given the fact that the technology is reporting on their physical movements and whereabouts. Even

[131] Members did not feel it is beautiful enough to share how it looks like with the general public.

though most hackers could handle and often prefer obscure user interfaces such as terminal commands, they felt important to make the interface so obvious that nobody could miss it. The often cited use case – which I rarely observed during the field work – was that sometimes members would want to come to the hackerspace *without opening it*. On the one hand, sometimes they want to concentrate on a project without risking that any visitors would stumble in the space, because if that happens then they feel compelled to give a guided tour and any help which may be needed. On the other hand, sometimes they would simply pop in for a minute in order to fetch something but would not stick around to keep the space open. In the latter case an automatic sensor could be misleading and disappointing if somebody notices that the space is open, heads down to the space and then finds it empty, thus closed. In my view these excuses are less practical and more ideological, highlighting the rights and responsibilities of members. Members have a right to understand, control and modify the technology that they are required to use, while they also have the responsibility to share the space and the technological expertise with any potential user who happens to be around. Once again, technological productivity, knowledge production and education are so entangled with each other in the hackerspace that they become an integral part of the same sociality. Indeed, these rights and responsibilities are not necessarily articulated in discourse but simply set in stone, e.g. encoded in the technical systems that frame social interactions.

Third, given the above, the absence of a haptic interface is a way to read the particularity of the local engineering culture. The door system in fact has a web interface superficially similar to home routers where members can log in and for instance make sure that the door system does not automatically set the space state to open when they only came to pick up something they left there. This is possible because there they implemented a delay between the door system learning about members associating with the wireless access point and triggering the change in the space state. Since Bitlair members worked on network protocols in many ways, a web interface may feel as intuitive to them as a button. In fact BitLair have already featured in section 6.3.1.3 noting that they supplied programmable networked lighting for the OHM hacker camp in 2013 and volunteered for the Network Operations Centre (NOC). Moreover, Bitlair received much media attention and some notoriety in the hacker scene for their reverse engineering efforts described in Hofman (2013) and analysed in Aibar and Maxigas (2014b). After the hackerspace moved into a large barn equipped with a networked alarm system speaking a proprietary (and therefore effectively secret) protocol, Wilco Baan Hofman and others deciphered it so that they can communicate with the security system via IRC (Internet Relay Chat). They also found vulnerabilities in the process and reported the problems to the company which did not react, so that the hackers escalated their findings to the agency responsible for government security that made use of similar systems. Finally, a special task force was formed in the state security organisation and the company was forced to implement the modifications proposed by the hackers.

As evident from the above descriptions, there is a lot going on in the hackerspace at Amersfoort about networking, often mixing the explorations of protocols with their effects on offline infrastructures.

Therefore, *the most prominent activities at Bitlair are arguably about hacking network protocols in an eminently techno-social sense, constructing and structuring social relations by technical means. Understanding, documenting and proposing protocols is at the heart of Bitlair's activities like the door system, the SpaceFed shared authentication system and the SpaceAPI for sharing statistics between hackerspaces are both technical achievements that expand technological infrastructures and social contributions to building a more tightly networked hackerspaces community. While projects and people at H.A.C.K. are showing a way in proper design and implementation which have characteristics of craftsmanship, Bitlair's contributions are more pragmatic yet managerial because they propose, advocate and maintain infrastructures for the hackerspaces community – perhaps somewhat like civil engineers.

10.1.2 London Hackspace, London, United Kingdom

The door system at the London Hackspace is interesting because it is one of the few door systems where a historical record of its various stages of development is easily obtained. These help to systematise the observations from door systems which are geographically wide-spread, technically diverse, and found at the most different stages of their lives. Even though current data is insufficient for constructing a general technological trajectory of door systems, a few stabs can be taken in that direction – stabs which can reach essential points. The most essential one is a widely recognised assumption in historical anthropology and archaeology: the artefacts of a culture on the rise are increasing in complexity over time. In line with such an assumption, I argue that viewed from the present historical horizon, door systems in hackerspaces tend to expand with each year of their operation, even if they see smaller setbacks.

The initial version of the door system in London was one of the first infrastructures installed at the hackerspace, prompted by people getting closed in because the space was big and the keys were still few (see Figures 16 and 17). In 2010 the door system tradition was already afoot between hackerspaces, so the spinner wheel could be taken as an ironic comment, a parody of techno-fetishism, and a demonstration that it is not the technology but the idea that counts. The simple paper contraption – a spinning wheel – speaks to several working hypotheses built up until now at once. Firstly, that the door system is a widely recognised idea in this particular milieu. Second, that it is considered an essential part of hackerspaces which have to be implemented as soon as possible. Third, that the OSHW ethos points beyond itself to a realm of social practices, social relations and socialised expertise where questions of licencing are nothing but an afterthought, and even sharing the schematics is not essential. *Within* a social group where expertise is widely available and production infrastructure is

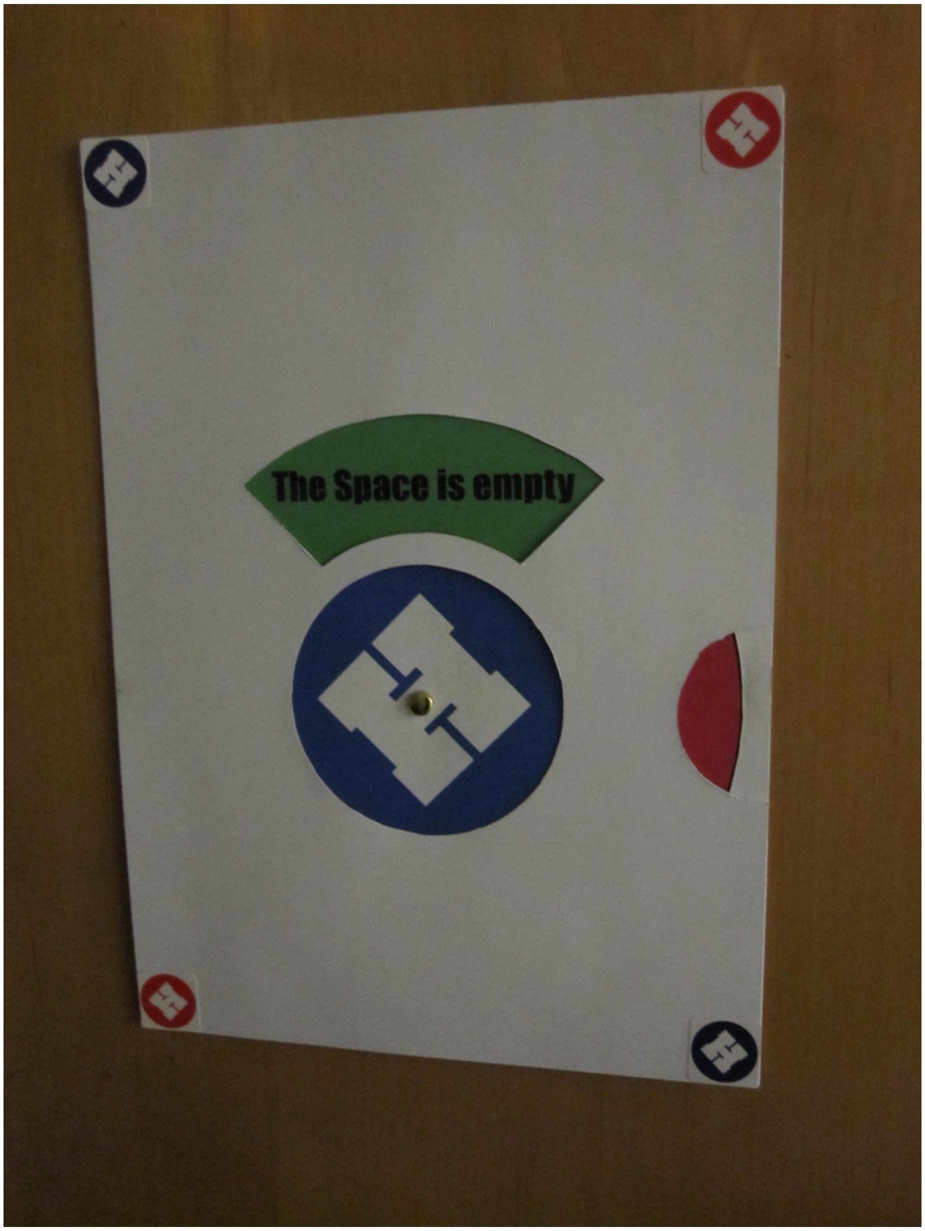


Figure 15: London Hackspace door system, early version, in use. Licence: CC-BY-SA 2.0. Author: Charles Yarnold. Source: https://www.flickr.com/photos/solexious/4517613541/in/pool-londonhackspace

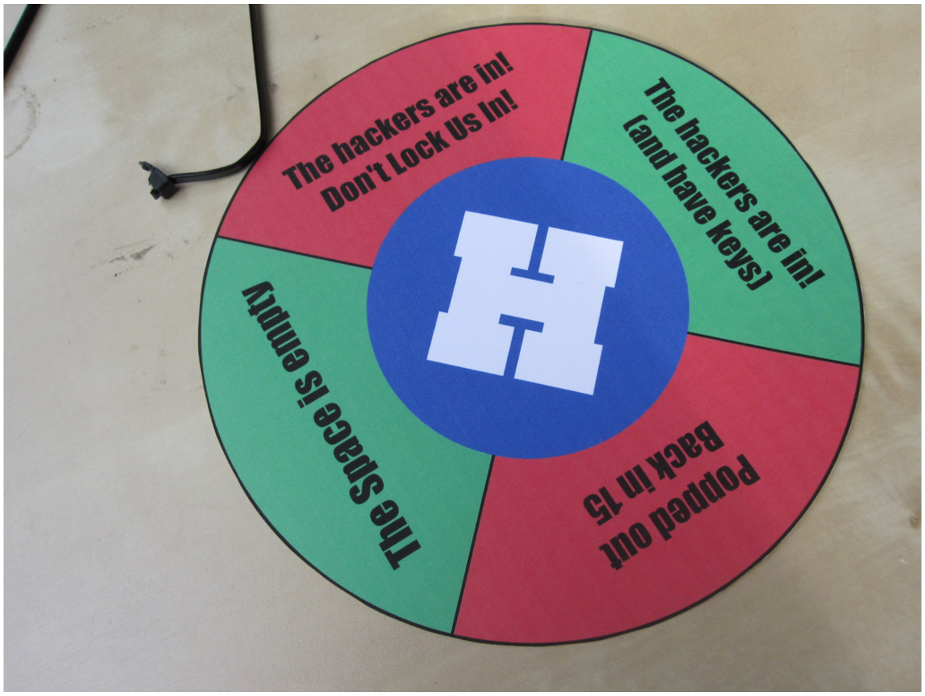
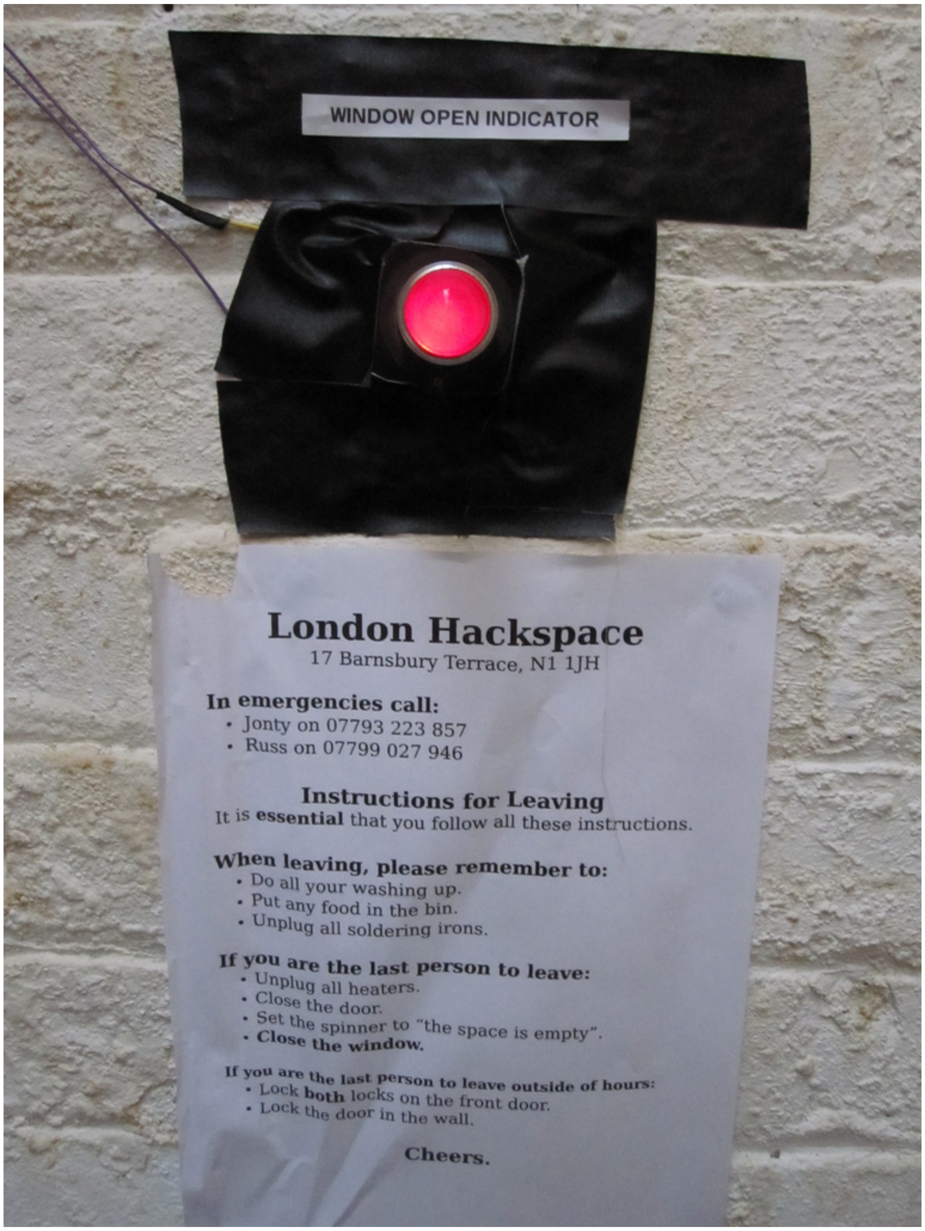


Figure 16: London Hackspace door system, early version, in use. Licence: CC-BY-SA 2.0. Author: Charles Yarnold. Source: https://www.flickr.com/photos/solexious/4517613537/in/pool-londonhackspace



London Hackspace

17 Barnsbury Terrace, N1 1JH

In emergencies call:

- Jonty on 07793 223 857
- Russ on 07799 027 946

Instructions for Leaving

It is **essential** that you follow all these instructions.

When leaving, please remember to:

- Do all your washing up.
- Put any food in the bin.
- Unplug all soldering irons.

If you are the last person to leave:

- Unplug all heaters.
- Close the door.
- Set the spinner to "the space is empty".
- **Close the window.**

If you are the last person to leave outside of hours:

- Lock **both** locks on the front door.
- Lock the door in the wall.

Cheers.

Figure 17: London Hackspace, window open indicator and human protocol notes. Licence: CC-BY-SA 2.0. Author: Charles Yarnold. Source: https://www.flickr.com/photos/solexious/4717743981/

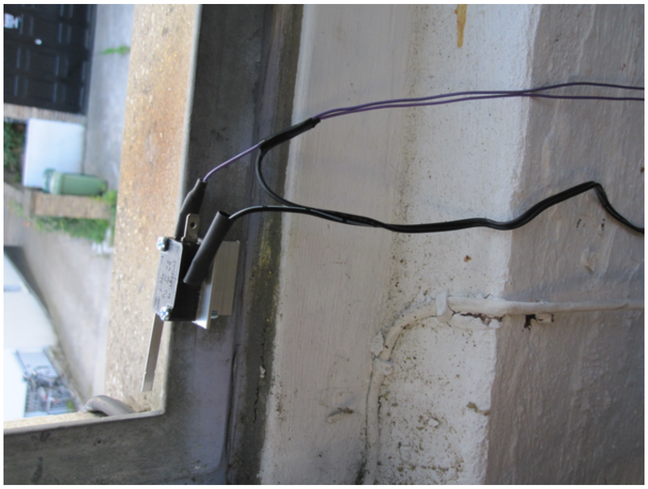


Figure 18: London Hackspace, window open indicator, sensor. Licence: CC-BY-SA 2.0. Author: Charles Yarnold. Source: https://www.flickr.com/photos/solexious/4717743981/

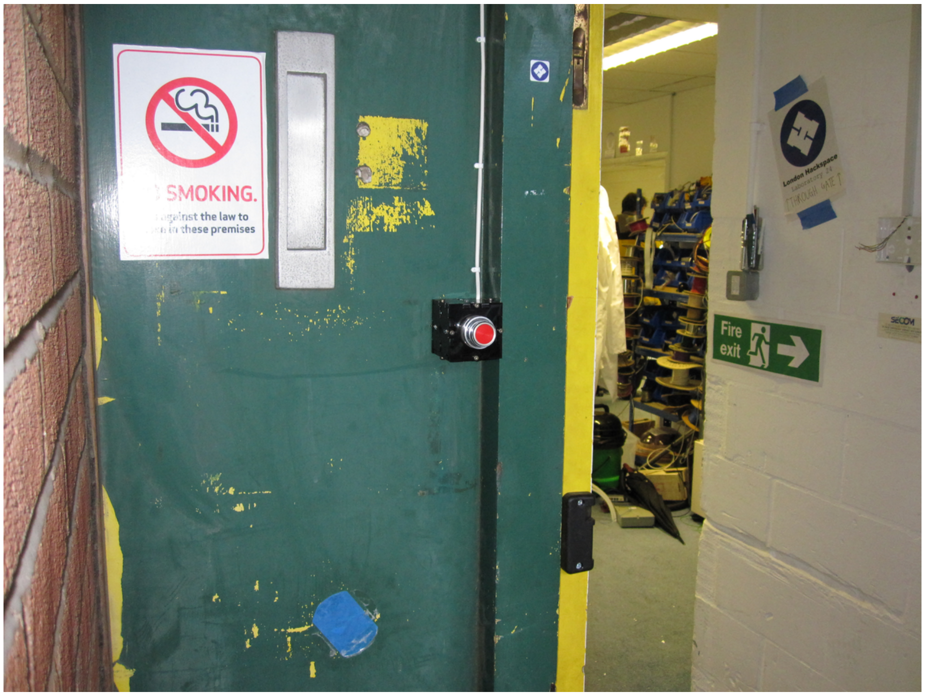


Figure 19: London Hackspace bell in context.

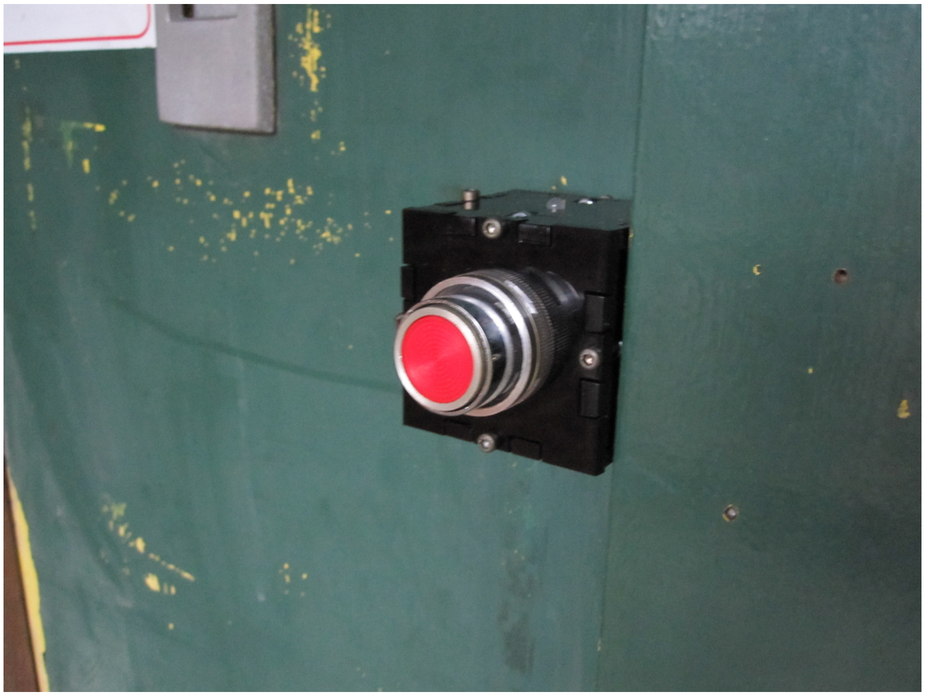


Figure 20: London Hackspace bell installed.

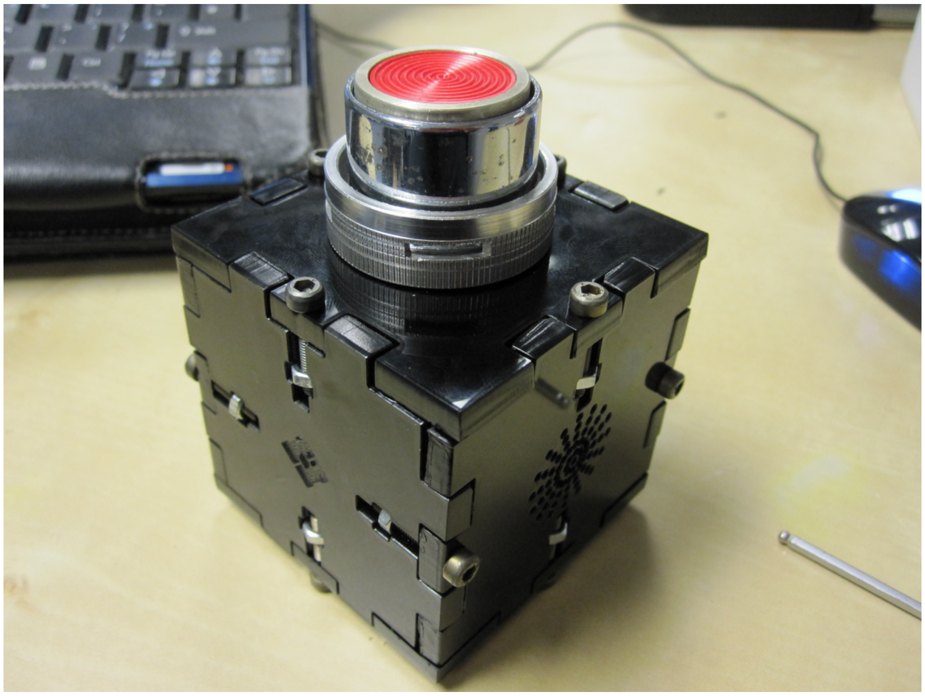


Figure 21: London Hackspace bell, close up.

widely accessible what counts is the sheer idea. The ultimate irony of the paper contraption is, however, is that its purpose and functionality is diametrically opposed to the raison d'être of door systems. While classic door systems were designed to let people on the Internet know when the space is open so that they can come in, the paper contraption was made to let people in front of the door know that the space is open *in order to prevent them from closing it and go home*. In other words, while most door systems target primarily the visitors without keys who want to go to the hackerspace, the paper contraption targets key holding members who want to go away from the hackerspace. In fact the very reason it can serve as a low-tech analogue commentary on the high tech digital obsession of hackers (what Wyatt (2008) calls the digital imperative) is because it does not need telecommunications because its audience is exactly the people in front of the door.

As the hackerspace grew, the door system became more and more elaborate. First, it was just a paper slip in an envelope, that evolved into a fancy door bell. The next version could already tell people outside if the hackerspace is open or not – the basic functionality of door systems as I defined them – and the last version identifies attendant members personally. By now it crafts elaborate statistics about the number of people in the space, the amount of electricity being used and simply entering the space sets in motion a sequence of unpredictable events reminiscent of a Tom and Jerry cartoon.

Today, the London Hackspace is the second most populous hackerspace in Europe, rivalled only by the Ur-hackerspace c-base in Berlin, Germany. As the number of members grew it also became more diverse and now it seems to me the least privacy conscious of the hackerspaces on my radar. Paranoia about privacy is prevalent in most European hackerspaces, but not in London. This could be related to the fact that London is widely reported to be the surveillance capital of the world. A European Union funded research project in 2004 coordinated by the Technical University Berlin's Centre for Technology and Society pronounced that "London is currently the unrivalled world capital for CCTV [Closed-circuit television] in public streets and places." (Hempel and Töpfer 2004, 28) By 2010 when the paper contraption was made, a widely cited industry report claimed that the United Kingdom has one CCTV per every 11 citizens, and most are operated by private entities (Hronesova, Caulfield, and Guasti 2014, 11). Nowadays the most prominent feature of the door system is probably the electronic voice announcing visitors (used in HSBXL in the same way, even though implemented completely differently). It is a defining experience of hanging out in the hackerspace and inevitably gets into reports of hackerspace tourists, such as Treb0r (2011):

> We hung out for two hours or so, and it was interesting to watch various people come and go – each announced by their personal sound sample or robotic voice when they swiped into the space with their RFID oyster cards.

These arrangements mean that members entering and leaving the space are clearly identified, both in electronic logs and to the audio audience present. The former happens through RFID cards that members have to register with the electronic roster of the hackerspace and touch to the card reader at the door. Even though this technology is held to be fundamentally insecure by hackers, an increasing number of hackerspaces rely on them: the common argument is that in case somebody defeats the RFID security system and gains unauthorised access, then they proved themselves to be real hackers and therefore they are more than welcome to the hackerspace. In line with such line of reasoning, members can register any RFID card for opening the door, for instance an Oyster travel card that is a staple of life in the UK capital.

Members have to give their legal names when signing up with the London Hackspace Foundation, and their full name will be announced by the robotic voice when they enter – unless they go the extra mile to configure another text message or sound sample they want to be played for announcing their arrival. The configuration lends a theatrical effect to entering the hackerspace and a sense of nobility to members because their entry is announced by a machine, like the "French butler" contraption discussed in Latour (1988b). Of course, hackers from smaller hackerspaces who are more privacy conscious would object to their personal movements identified, recorded, stored and broadcasted along with their names. Even if these measures can be overcome, the default options are not very respectful for individual rights. I guess that the diversity of membership in the London Hackspace have drenched these voices. Indeed, most people frequenting the space were happy with these arrangements during the time I have spent there. At the end of the day a space with a thousand people coming and going is akin to a public location where the fact that someone was there proves little in a hypothetical criminal investigation. On the other hand, it may be useful information to keep track for the community: not just for the odd moment when something gets stolen but also to stay connected and be able to know how to get hold of people.

However, I think that the most useful aspect of both the RFID and speech synthesizer usage is to negotiate the difficulties presented by running a hackerspace with such an unusually large number of membership. The former makes it easy to sign up members and the latter makes it easier to know who is who. Both practices can be interpreted as a technologically enabled way to intervene in the dynamics of the community and to adapt to the challenges presented by a community space for flexible workers. More than any laboratory or community space, London Hackspace is open practically non-stop, and people living precarious lives are coming and going without loosing the feeling that the hackerspace is their second home and the sense of familiarity with other members. It can be argued that London Hackspace members used technological means to address the social problems arising from the influx of members. Later on I will address the question of access control as an extension of door systems in section 10.4.2, and the problematics of translations between human and machinistic agency in section 10.5.

Many members also mean many people who want to hack at the door system. The door system by now grew around the whole expanse of the space and incorporated many machines and projects operating in the room that originally served independent purposes. In fact by now no one member of the hackerspace can recount all the things that happen across the technosocial spectrum when somebody turns up at the door. The wiki page of the "door control system" project is an attempt by the community to draw these lines together, but the description makes so many references to other devices and projects that it is hard to make heads and tails of it without an intimate knowledge of the London Hackspace universe:[132]

> There are listeners on Babbage that connect to robonaut to announce on IRC, and ~~flash the lights using Lighted~~. By default, this will include your full real name. If you wish to change this behaviour, you can set up a nickname in the cards section of the member area.
>
> ~~hamming also runs listeners for the scrolling led board and the audio announcements.~~
>
> The announcement listener uses the GLaDOS voice. You can generate and use your own file as a greeting.
>
> The code that runs the bandwidth meter on tesla also listens for doorbell and member entry messages (London Hackspace contributors 2014).

Without going into specifics, I would like to highlight three points about the description. One, it is trying to share expertise by describing how the system works, not merely what it does. Contrast this with the manual for a piece of consumer electronics like an immersion blender: it will focus on what the artefact does rather than how it works – while here the former question is often left unanswered. We have no idea what gets logged about somebody entering the space but we do learn that we have to look into the "code that runs the bandwidth meter on tesla" (the latter a name of a computer, lower case by convention) to find out. Two, the description emphasises what users could do to change how the system works if they wish to change its behaviour, therefore encouraging participation in the refinement and development of the system. A blender manual would certainly not advise users about how to change the motor to a stronger one or make a different kind of blade assembly. The language is particularly empowering, telling the user that "you can" do lots of things. Three, that such mix of applied expertise and invited participation leads to a proliferation of functions and names that becomes hard to navigate and appears like an ever changing mangle to the reader. These can only work together as long as the design principles ingrained in the engineering aesthetics of unfinished artefacts are maintained.

[132] Parts of the text are crossed out in the original.

Steward (2014) explained on the mailing list of the hackerspace why they did not design the most elegant, minimalist and robust solution for the access control system that lets people in, enumerating the principles of reproducability, modularity, transparency and simplicity:

> The last thing you want with an access control system is to design it around one hero who is the only person able to maintain it (that's separate to onlyhaving one person making changes). Our decisions have prioritised:
>
> - how cheap and off-the-shelf it is
> - how easily parts can be swapped out
> - how easy it is to debug over the internet or by instructing someone who's never touched it before

Note that while hackers may agree on the ultimate goals, it is not trivial to decide if a bare microcontroller chip or an Arduino is a more "off-the-shelf" component, nor which one is ultimately cheaper and more replaceable. If it is easy to understand, modify and debug something also depends on the particular skill sets of participants. Members of the hackerspace in Budapest would perhaps argue for the opposite technical solutions for achieving the same design goals. However, the centrality of some concepts still stands out across hackerspaces, as well as the geographically specific local interpretations of these concepts by the London Hackspace tradition of engineering.

Therefore, *door systems arguably help London Hackspace members to keep track of themselves and their community through employing technical means for bringing and keeping people together, and in the same gesture opening these means to them as an invitation.* As door systems grow and membership numbers increase, it becomes non-trivial to exercise expertise while maintaining participation. Here the thrust of hackership has little to do with craftsmanship – it is more about implementing your ideas in cooperation with others so that they fit into existing and path dependent infrastructures, all the while explaining them to your peers – that is to the hackerspace membership.

10.1.3 HSBXL, Brussels, Belgium

The HSBXL door system shares many features with the one at London Hackspace, like RFID access control and computer voiced announcements. Since the hackerspace has around 30 members and only a few are interested to get their hands dirty with elaborating the door system, it is much more of a personal project for askarel, a heating, ventilating, and air conditioning engineer by day, Linux sysadmin and hardware hacker in the after-hours. His curriculum on the LinkedIn business-oriented social networking service lists The Black Knight as his only public project, the credits shared with four other hackers. The Black Knight is designated as an “RFID-based Access control system for Hackerspace Brussels, with garbage day notifications.” (Pasteleurs 2013)

The Black Knight does not actually provide for the basic functionality of door systems according to my definition: this job is left to Pamela, jointly developed by Hackerspace Brussels and Whitespace Ghent,[133] according to the liner notes (sandb 2011). Similarly to the solution used in the Bitlair hackerspace in Amersfoort, The Netherlands (see 10.1.3), it uses the wireless access point as a sensor to know if members are in. However, the display is much more theatrical: the names of connected computers are floating around the logo of the hackerspace. In the case of HSBXL it is actually mathematician and cybernetics pioneer Robert Wiener wearing Mitch Altman’s brainmachine, the latter itself a nod to another cybernetic artist Brion Gysin’s own Dream Machine from the 1950s (Pickering 2010, 80–83 and 419). In this way the names of participants are not exposed, even though people who know the names of their computers can identify them. Moreover, the space state is decoupled from the actual door, and there is no button.

The Black Knight, on the other hand, manages the entrance door in many respects. One, it lets authorised members in, or anybody who rings the doorbell once it is put in “party mode”. The RFID identification draws its database directly from the payment system developed for the hackerspace, so that members who fail to pay their fees are automatically denied entry. Two, it makes sure that if there is no electricity, the electronic locks do not stay locked. A special feature is the *emergency box* installed near the entrance which opens the door if the glass on it is broken. Three, it advices any visitors through the robotic voice synthesizer to take the trash out on the relevant days. This is designed to be annoying enough so that members are motivated to do their chores. Four, the system advises if there is mail in the physical mailbox.[134]. Five, a tripwire is being implemented to detect if the door is not opened properly but rather broken in. Members say that the hackerspace is currently located in a relatively rough neighbourhood which calls for a strong door and dense security. HSBXL will soon move to another location in the city centre with more doors available and less caution necessary. The current build is really geared towards the one robust front door, so the system will have to be implemented in a different way, and while requirements and ideas change, some parts will be surely reused.

Interestingly, there is no documentation of the hardware layout. Askarel’s explanation is that it is quite easy to see from the published software code how to build the appropriate hardware for it. Indeed, some part numbers and GPIO pin numbers are documented in the code. However, my estimation is that it would be impossible to replicate the system given only the software. Instead, a person who sets out to replicate the HSBXL setup would have to reinvent a good part of it. It does not help that the software is licenced under the GPLv3 so it is FLOSS, but it would make such efforts at least legal. In practice what

[133] Another hackerspace, also in Belgium.

[134] I am not sure how because this has been implemented after my visit to HSBXL.

happens – and what is the main point I try to establish by presenting a panorama of different implementations – is that hackers find the general idea of a door systems compelling, but prefer to reinvent the wheel each time they open a new hackerspace or sometimes even when they move to a new location. So while replication is made hard by the scarce documentation, reinvention is certainly possible and in fact happens in practice. This is possible because door system are not difficult projects in terms of hardware hacking and as we have seen earlier, they can be built starting from a basic prototype to became more arcane as time goes by, the number of contributors and users grow, and members learn skills and think up new ideas. In other words, the door system can grow as a factor of the increase in participation and the increase in expertise over time.

One possible interpretation is that *the door system at the hackerspace of Brussels is not there to prove an abstract point in a historical perspective, nor to implement a system to be adopted by fellow hackers everywhere, not even to cultivate community – it is there to prove that hackers are free to disregard mainstream engineering standards and seek their own pleasures and interests in the pursuit of virtuoso engineering performances.* Having said that, the door system does respond to the immediate needs of the community in the "scratching an itch" way that Raymond (1999) popularised, whether it is physical security, nudging tenants to take out the trash, or the computer telling you that you received a paper mail. Even though many values of craftsmanship go into the system, tinkering may be a better paradigm for the free-wheeling experimentation and inconsiderate construction that leads to the ever expanding features of The Black Knight.

10.1.4 Hack42, Arnhem, The Netherlands

Members at the Utrecht hackerspace RandomData in the Netherlands told me that if I want to see door systems, I have to go to Hack42 in Arnhem. They were right. Hack42 has the most sophisticated door system I saw so far. Like most other wall-mounted contraptions in the space, the physical interface of the door system is built into a nondescript electric box, but unlike any other, it sports a red industrial handle (pictured in Figure 24). The original mains control knife switch that was installed at this WW2-era German barracks was found later and put in place of the handle. Reminiscent of a Frankenstein movie, all it does is to interrupt (cut/uncut) an Ethernet cable going from the box through the wall, where the labyrinthine electrical installation of the hackerspace converges, basically occupying its own room. The basic setup closely resembles the one in Dublin at the TOG hackerspace.

Then various things happen, including the usual announcement on IRC, website and Twitter, but also most lights automatically turn off in the building, as do many "safety power sockets" and "network controlled power sockets". Many conventional electric sockets as their circuits terminate next to the door system box which is wired up with the fuses in the central electric box. This ensures that

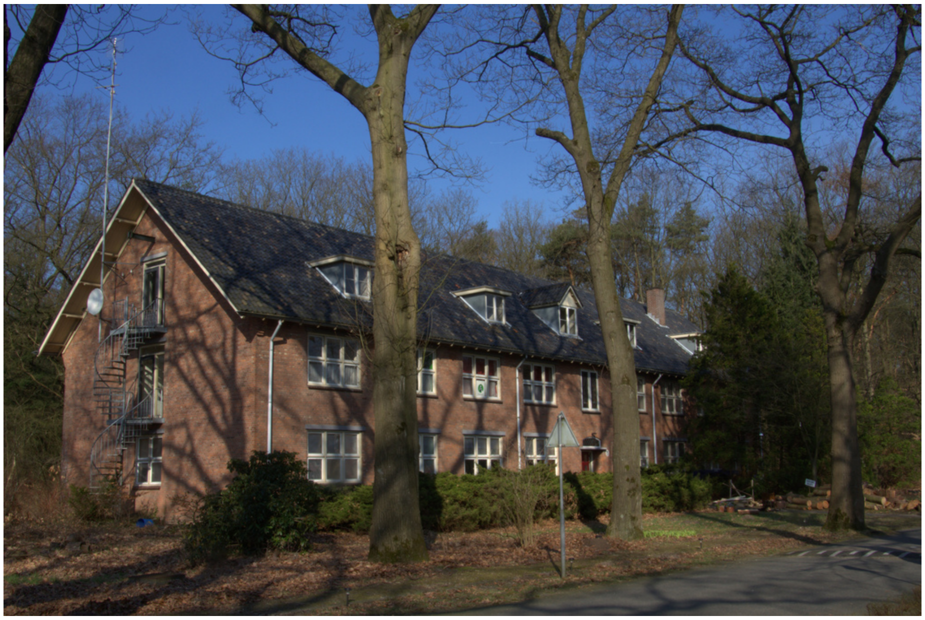


Figure 22: Building of the Hack42 hackerspace in Arnhem, The Netherlands. Licence: CC-BY-NC-SA 3.0 Unported Author: Stitch Source: https://hack42.nl/wiki/Bestand:Pand.jpg

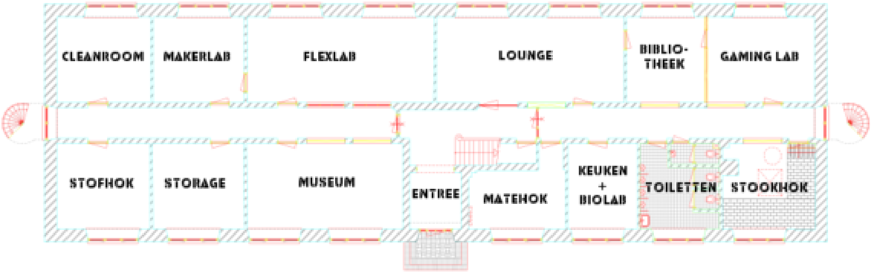


Figure 23: Ground floor plan of the Hack42 hackerspace in Arnhem, The Netherlands. Licence: CC-BY-NC-SA 3.0 Unported Author: Dvanzuijlekom Source: https://hack42.nl/wiki/Bestand:Plattegrond_kkn6.png

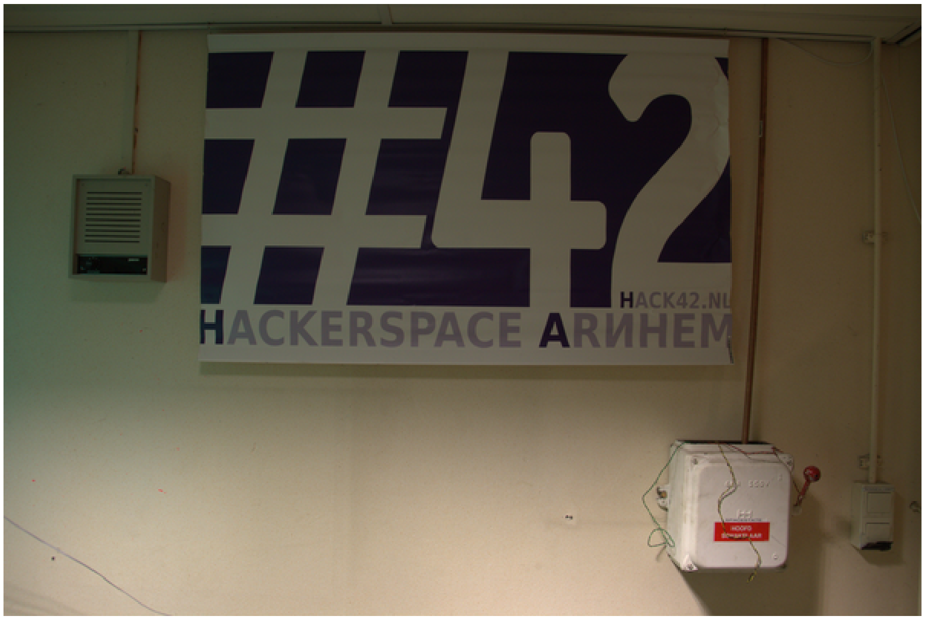


Figure 24: Door system switch, on the bottom right, Hack42 hackerspace in Arnhem, The Netherlands. Original legend reads (emphasis mine): “Hack42 Spacestate switch. Flip the switch to the ’on’ position and **the entire building will come to life**. The ’open’ spacestate will be twittered, displayed on our website and the IRC channel.” Licence: CC-BY-SA 2.0 Unported. Author: Dennis van Zuijlekom. Source: https://www.flickr.com/photos/dvanzuijlekom/6556630813/in/pool-hack42

for example the soldering irons are not left on when members leave the building, neither the sensitive vintage computers in the museum room, nor the electric oven in the biohacking area. Since there is also a building-wide telephone and public address system, there are plans for making a recorded voice announcement and activating the answerphone. The hackers explaining the system mainly framed these features health and safety issues, and of course as a matter of convenience. The old barracks has several floors and numerous rooms. All in all it is more than 11000 square meters of flotsam. The complex is located in the outskirts of Arnhem which makes it very inconvenient to go back for turning a light or a laser cutter off in case members forgot to do it before they left, and the size of the building means it takes much time to check each room before leaving.

The same single board computer at the heart of the door system is also gathering statistics from the hackerspace. The knife switch state is used to graph the time the hackerspace is open or closed, and the fuse box is equipped with a custom electricity meter installed next to the official one. The former meter takes more precise and more frequent measures of electricity consumption which are also plotted on the website. These values are used as a rough estimate of hacking going on in the space. Their counterpart is the other data stream coming in from the hackerspace's bar computer: the amount of mate consumed. Hack42 have already been mentioned as a unique place in the world where you can get forty two different takes on the original Club Mate drink.

Finally, almost all rooms are fitted with a standard single wire temperature meter. As before, their values are also turned into graphs. These are important because it is difficult to heat all the space during the winter. The first machine that was restored when the hackerspace moved there is a 50KW monster heater that was left over from the war, which can make as much as 30℃ in the lounge during dreaded Dutch winters. However, radiators that the heater feeds around the building are not as effective: utility water pipes could still freeze when it is under 20℃ for a few weeks, and the laser room needs extra electric heaters to keep the gear safe. Therefore, it is nice for members to know that they will arrive to a warm place before they depart to the outskirts to join their kin, or to know that their favourite tools are not in danger of cooling out when they are not there. While many other hackerspaces monitor the temperature just for fun, at Hack42 it is easier to argue that it actually makes sense.

As in Bitlair, there is a decent delay between toggling the physical switch and toggling the actual space state in the form of announcements and database writes, so that fascinated visitors who turn the enormous switch on and off several times in quick succession do not create too much noise online. Still, every turn of the switch as well as various infrastructure events are reported through the restored loudspeaker system throughout the complex. The same system can be used to place phone calls to any room of the hackerspace or to outside world normal phone numbers. Vintage phones from the most disparate eras are placed in each room, becoming a museological collection of their own, complementing the collection of more or less working computers, projectors, cameras, and typewriters that still see regular use during the sessions of the “typewriter society”. Along with the historical location of the building itself and its restored military heating, electric and communication systems, these artefacts provide an almost antique backdrop to the high tech computing culture that is still the foreground of activities. In this vein it is noticeable that over time the door system starting at the entrance came to encompass the whole building through wires, signals and networks, even growing beyond its walls to web servers, chat servers and social networks. What started as a simple device for signalling presence came to be an elaborate system of control and feedback. In the same way as members restore, use and improve old cameras, typewriters and computers, they also restored, used and improved the building itself and its technological infrastructures that date from the era when computers were first developed. Technological eras thus blend, fuse and stratify in the hackerspace into a single time-agnostic vision of engineering practice concerned with the care for humans and non-humans.

One possible interpretation is that *the door system of Hack42 is a safety net which makes sure the hackerspace opened and closed according to the proper procedures, and people inside the building have a sense of what is going on.* In other words, pulling the Hack42 door system switch puts you in a peaceful state of mind, knowing that all is safely deactivated. Once again, instead of craftsmanship we see another attunement for engineering works here, which may be best captured with the job descriptions of caretaking, maintenance and reparations.

10.1.5 TechInc, Amsterdam, The Netherlands

TechInc is one of the newest hackerspaces in The Netherlands, founded in 2011. Therefore their name for the door system already references the SpaceAPI, the initiative to have a standard interface for querying space states between hackerspaces: it is called the SpaceAPI button. It is a very clean and simple implementation, evidently based on a good overview of other solutions in nearby hackerspaces, where many of the members are involved in tandem with TechInc. As shown in Figure 26, the button is a standard one for industrial applications, interrupting a circuit in the Nanode single board computer mounted right below it in a plastic box. The Nanode is a close relative of the Arduino, but it is geared towards networked (Internet of Things) applications – but more importantly, it has been developed in the London Hackspace, with which TechInc also shares members. Using a Nanode instead of an Arduino makes it easier to access the network, that is to reach the web server where the space state is logged. The contraption is as simple as pressing the button to change the state, and visual feedback is provided by the TechInc logo mounted above the button.

The logo of Technologia Incognita looks like the steering wheel of a classic ship, but it is actually also a working circuit board design which exhibits a common trick of that trade (Figure 27). So the logo on the SpaceAPI button

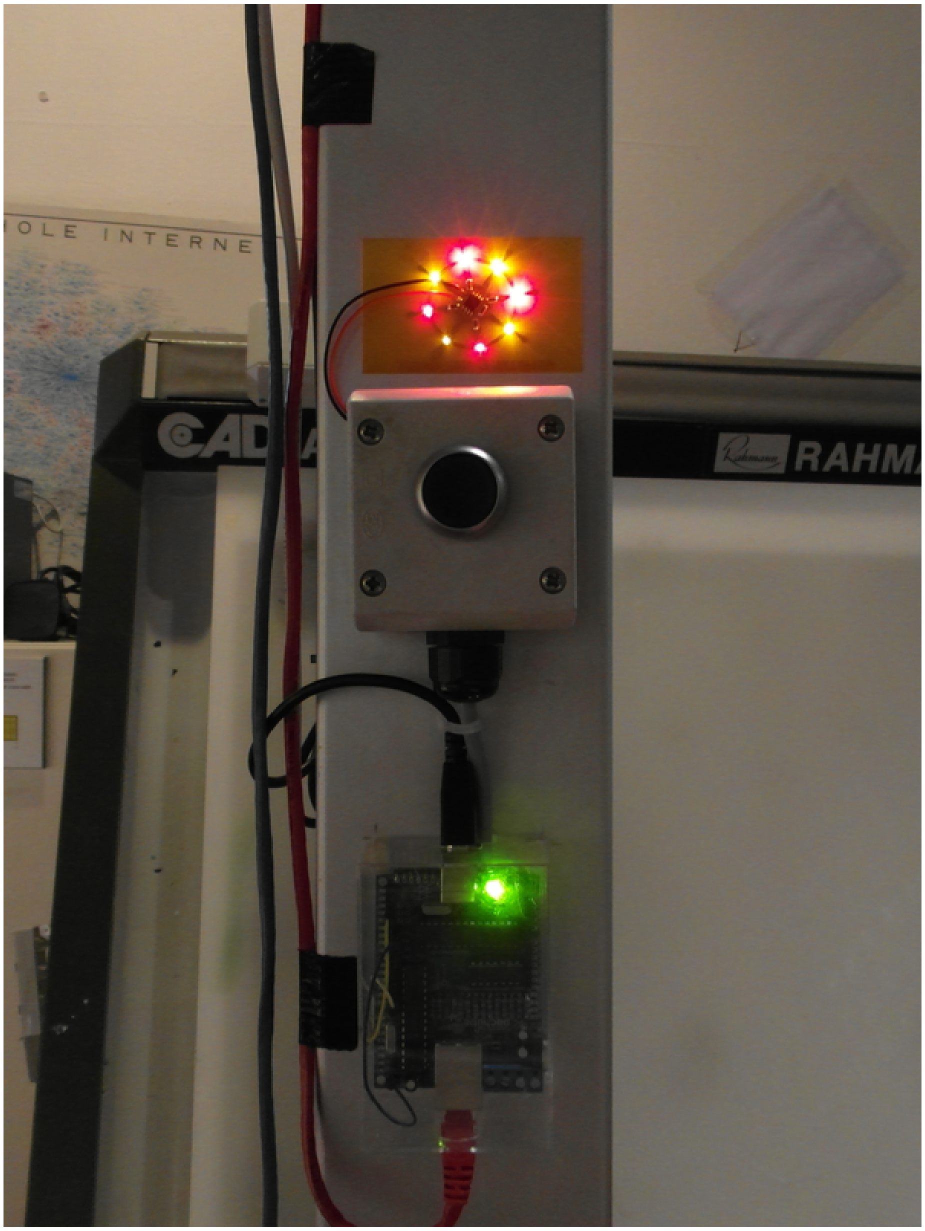


Figure 25: TechInc SpaceAPI button. Licence: All Rights Reserved. Used with the permission of the author. Author: Brainsmoke. Source: https://wiki.techinc.nl/index.php/File:Working_logo_badge.jpg

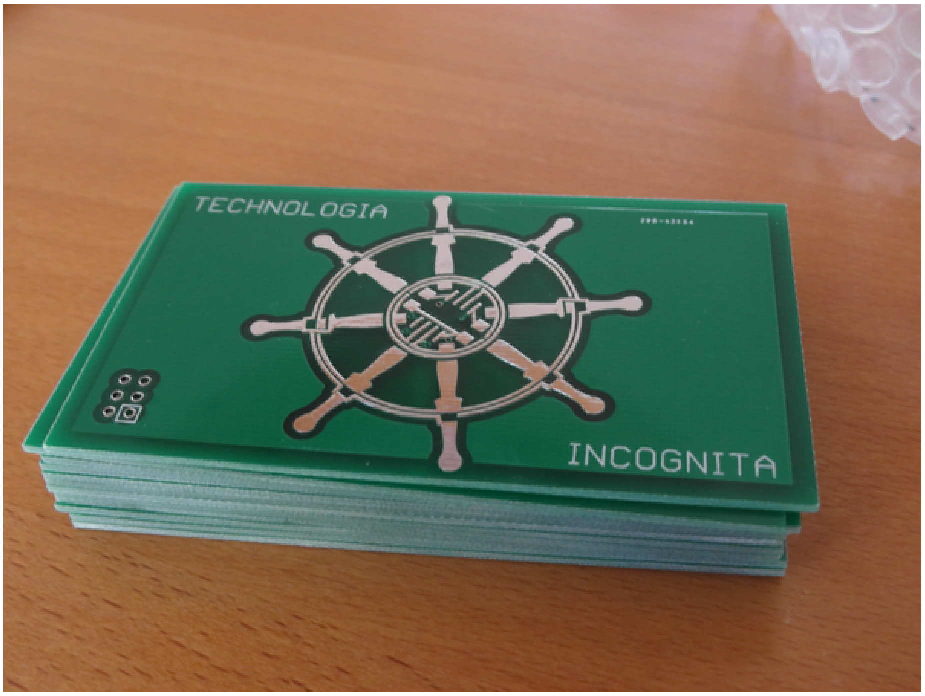


Figure 26: Etched TechInc logo PCB, ordered from factory. Licence: All Rights Reserved. Used with the permission of the author. Author: Brainsmoke. Source: https://wiki.techinc.nl/index.php/File:Test_batch.jpg

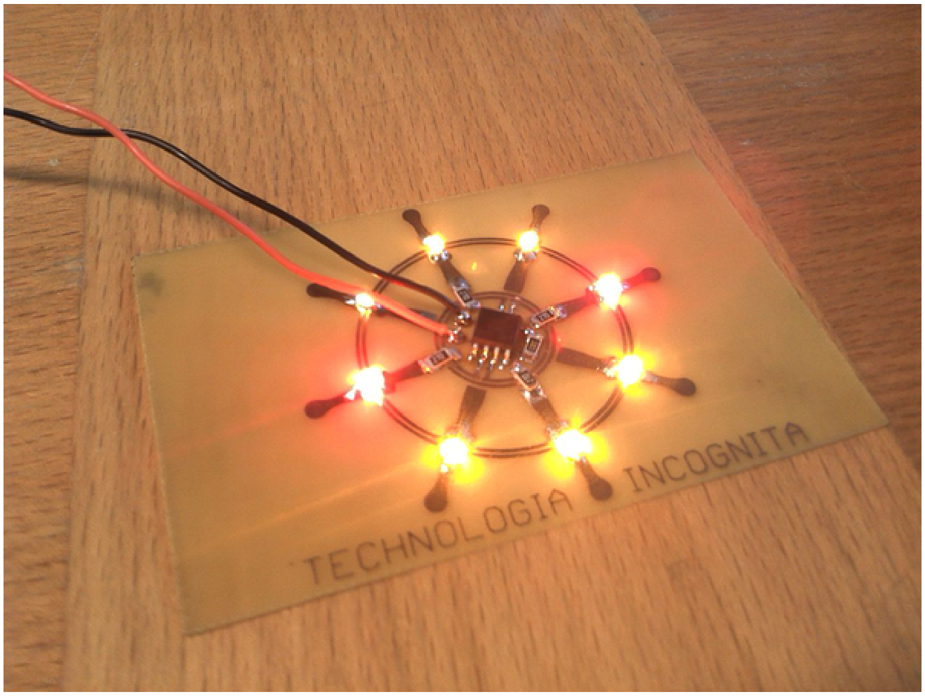


Figure 27: TechInc logo PCB, powered up. Licence: All Rights Reserved. Used with the permission of the author. Author: Brainsmoke. Source: https://wiki.techinc.nl/index.php/File:Working_logo_badge.jpg

is a small business card size circuit board with glittering chrome and shining LEDs (Figure 28), which lights up when the door system is activated. Finally, the logo blinking at various speeds signals one of five error states. Space state is logged and published through the aforementioned SpaceAPI, displayed as an OPEN/CLOSED sign on the website of the hackerspace and there is even an Android smart phone application which alerts its users on space state changes.

While the implementation of the door system seems simple and straightforward because it sticks to the basic idea of updating the space state, this leaves members who are closing the space with a list of tasks (Ultratux 2015):

> Here's how to close up the space when you're leaving:
>
> - Check if there's enough Mate (and other beverages) in the fridge. If not, please restock
> - Turn off space exhaust fan switch in the dirty room next to the entrance.
> - It's labeled "AIR FAN". Watch out; there are more switches there.
> - Turn off powerbar-controlled items. You can do that via either webinterface Powerbar or via Bash_the_Lights. Please confirm everything shuts off.
> - A few exceptions to this: kitchen Led bar light has a separate switch.
> - Turn off the LED wall/fireplace with the switch on the left side.
> - Shut down the arcade game console by pressing the "return coin" button.
> - Press the space button (if it's lit, and it should be lit, otherwise forgettaboutit)
> - Get out
> - Close door by pulling it shut.
> - If any rubbish is sitting in the hallway, please take it out

It is easy to see that even though some of the tasks have been already automated – for instance through a centralised power bar and light switch – there are almost a dozen manual steps to take. There are a couple of things to observe here. First, it is arbitrary which tasks are the responsibility of humans and which tasks are the responsibility of non-humans. Humans can build an automation mechanism for any of these which translates human agency to non-human agency. Second, menial tasks have been automated in other places and are in the process of being automated at TechInc too. Hackerspaces are complicated and door systems are a way for hackers to deal with such complications through offloading part of the complications to machines, which effectively means of encoding them into software and hardware. However, the particular concept of technology that hackers cultivate through engineering subcultures and put into practice through building unfinished artefacts in a way preserves human agency at the same time as fixing it into technological systems. Even though a menial task is automated, human agency is not ultimately shifted out of the scene.

There is some sense – manifested in the inclusion of buttons instead of sensors – of humans staying in control. There is a degree to which technologies remain transparent to their users: manifested in transparent containers for circuits, naked PCBs and dangling wires that are not hidden behind plaster. Furthermore, there is an attempt to explain how the system works, not just what it is doing, since there is always (a perpetually out-of-date) wiki page that documents the residue of technological creativity. Finally, these factors come together in the ability of the user who is empowered to intervene to modify the functionality of the artefacts. As a result, humans are freed from the burdens of labour *without* loosing the agency over influencing the material artefacts that structure their lives – a very particular vision and practice of technology. Of course, such a vision of technology is only operational as a practice as long as it functions in a context where the milieu is comprised of privileged workers with free time on their hands to learn and care about technology, ones who can raise their general intellect at the same time as they raise the complexity of their technology. As the descriptions here testify, the level of technologies applied in hackerspaces is in fact not very high. I have already gone to considerable length to demonstrate that there is in fact a strong cultural and normative tendency in the hacker scene to keep technologies simple. Using old technologies is as much part of these efforts as building new technologies that are more easily grasped by their users.

At a hackerspaces roundup discussion in a hackers convention (Becha 2012) TechInc members choose to highlight a LED display they have built. As they emphasised, the project is very ordinary in terms of what people usually do in hackerspaces, but its significance for them was that it was built by programmers who did not work with electronics before. In the same manner the SpaceAPI button and the logo PCB allows members and visitors to get familiar with hardware hacking.

The logo itself is available as a kit – a form of delivering the project that is between the full informality of the unfinished artefacts described here and the market circulation of commodities. Some 3D printers and other electronics products are sold as kits in a way that is practically equivalent to IKEA's flat pack furniture concept. However, kits in hackerspaces rarely reach the market, and they are usually available only on site and at events for little more than the cost of their parts. Their point is to give visitors and members something simple to hack on, or to provide solutions to common problems that hackers have, even though they may play some small part in contributing to the financial sustainability of the hackerspace too. Many kits focus on initiation (which will be discussed in section 10.2.1 more length): that is to give a positive experience in hands-on electronics works to newcomers, who can produce a concrete artefact testifying to their soldering and assembly skills.

TechInc embarked on the way to become a hackerspace and integrate into the scene – driven by the work of `brainsmoke` and other members who were already familiar with other hackerspaces – in three ways. First, through beginning to produce functional small scale electronic artefacts like the PCB logo kit or the door system itself, which are the staple products of hackerspaces. Second, by automating maintenance tasks of the space itself through small scale electronic artefacts, in order to transform the rooms into place where humans and non-humans cooperate according to a specific conception of technology. Third, by publishing information about the space state through the SpaceAPI which integrates the various implementations of door systems under a coherent umbrella, enabling the public to visit when the space is open and technically putting TechInc on the map of hackerspaces.

One possible interpretation is that *the door system of TechInc is a ceremonial artefact which signifies their entry to the community of hackerspaces through the connection to the SpaceAPI.* Therefore, the door system is a *performative sign* of belonging to the community of hardware hackers. In terms of craftsmanship the TechInc story is reminiscent of displaying a "master work" in order to enter the community of qualified craftsmen.

10.1.6 Conclusions about the diversity of implementations

The panorama of door systems presented here serves as the primary empirical basis for further analysis in the subsequent sections. However, a couple of basic analytical points have been already established. First, door systems are culturally specific to hackerspaces. The mere fact that there is a door system implementation in a shared machine workshop is enough to distinguish it as a hackerspace in contrast to hacklabs, Fab Labs, Makerspaces or TechShops. Second, implementations of door systems are highly diverse across hackerspaces, showing that it is more a genre of small scale electronic artefacts than a concrete open hardware project. The proliferation of door systems cannot be understood in reference to licencing schemes (as in FLOSS/OSHW), good documentation (as in a knowledge commons), or even ICTs themselves that enable collaboration (as in technological determinist interpretations of peer production like Shirky (2008)): it has to be accounted for directly in terms of social relations conductive to peer production practices – the specificity of the hackerspaces milieu. Third, door system implementations are geographically[135] and historically[136] specific. It is hard to make a systematic evaluation of the many factors involved, but I have shown examples big and small. On the one hand, in Dublin the city hosts major media monopolies largely because of national tax policies, which can be associated with the high number of professional software developers and system administrators in the membership of the local hackerspace, which paradoxically swings activity towards hardware hacking. On the other hand in Brussels the location of the hackerspace in a neighbourhood that is considered rough by some

[135] Depending on the members, the building and the city.

[136] Depending on the age of the hackerspace and the rise/fall of activities.

members gives a good excuse for more or less a single person (askarel) developing a sophisticated, robust, yet highly idiosyncratic door system whose software is written in an almost forgotten programming language (Pascal). Between these extremes, cases mostly highlighted members of a hackerspace cultivating a specific flavour of engineering culture that emphasises different aspects of hackerdom, so that the specific socio-technical values and design principles are encoded in local door system implementations. As archaeological findings, these artefacts then serve to preserve and perpetuate these local takes on hacking as a practice.

What does the diversity of implementations tell us about the way collaboration around unfinished artefacts happens in the hackerspaces? Internally, that is in the context of a specific hackerspace, door systems are a privileged centre of activity and cooperation between members. Hackers who come to hang out in the hackerspace eventually want to hack on something as part of the specific sociality associated with the milieu. In case there is nothing concrete that is being done or needs to be done in the space, the default option is to look around and hack on the infrastructure of the hackerspace itself – at the centre of which is the door system. Alternatively, it is not boredom but inconvenience which drives hackers to work on the door system. Especially in the first period that follows moving into a new building, there are many practical problems posed by sharing a space with non-stop access between a few dozen or more participants that want to coordinate their movements in order to socialise and collaborate with each other. The London Hackspace example where members are accidentally locked in the building is an extreme example – usually the inverse happens, e.g. that members cannot get in because no key holders are there or even because the door system access control component broke down due to a crashed computer – another episode that happened in the London Hackspace. In the next term, problems move from the perimeter of the hackerspace towards the internals. Windows left open in London, laser cutters on fire due to nobody watching them in Arnhem, garbage piling up in Brussels, or the alarm going off in Amersfoort are some of the examples of annoyances that hacker addressed by "scratching an itch" (Raymond 1999), that is, developing their locally specific small scale electronic artefacts to offload chores and responsibilities. In the final analysis, building a hackerspace includes hacking spaces through door systems.

Externally, the door system as a genre evolves through copying rather than collaboration per se. However, copying does not mean the reproduction of a door system implemented in one hackerspace in another hackerspace. Only specific features are copied and even when they are copied their are adapted to the local context, including the technical preferences of the implementers, the availability of components, the time available for the implementations, members' perceptions about the specific requirements of a building where the hackerspace is located, and first and foremost to the already existing infrastructure which introduces path dependence. Discussions around the next steps to develop a door systems in a hackerspace often make references to specific features of door systems in other hackerspaces. Importantly, these references are rarely picked up off the Internet initially. Members visit other hackerspaces and have first hand experience with alternative implementations. They meet at hacker conventions and even though hackerspaces do not bring their door systems to the site, there are discussions and demonstrations of door system implementation questions which influence local development. While online discussions can lead to specific developments in local door system implementations, it is typically Away From The Keyboard meetings between members of different hackerspaces which brings on changes. Once the inspiration is there, however, implementers look at the online documentation to learn about the way a specific problem has been solved in a specific hackerspace.

The discussion of door system implementations also serves to flesh out the convergences and divergences between unfinished artefacts and OSHW. As argued before, OSHW definitions cannot adequately capture the phenomena of small scale electronic hardware production in the hackerspaces because they are tied to legal instruments that make open hardware open. Unfinished artefacts, however, are defined by the social relations that produce them and which they produce in turn. They are unfinished because they never become an impenetrable black box or stabilise into a network that is resistant to intervention. Transparent or missing boxes, dangling wires and constant tinkering characterise door systems as unfinished artefacts. Despite appearances, these properties make them more functional, resilient and reliable in their specific environment. They can only exist in a specific milieu where participation in the community directly equals participation in the exercise of expertise in education, research and production. Despite the counterexamples below, at the moment a door system only makes sense for participants of a hackerspace: there are no other popular use cases in other contexts that I know about. Unfinished artefacts – including the r0ket badge presented in the previous chapter – therefore seem to be extremely tied to their social context and to a particular social group. Indeed, their fluidity and the agency of their users to study, modify and reproduce them basically disappears once they are considered outside the hackerspaces. That is because the hackerspaces provide the access to knowledge as well as the access to the material infrastructures that enable these freedoms. Hence, an unfinished artefact never stands alone but has to be considered in relation to its users, creators, maintainers and developers. Equally important are the relations of those people to each other and to the wider infrastructures they cultivate. Of course that is the case with any artefact, but such considerations apply to an increased degree to unfinished artefacts. Unfinished artefacts can be considered as the material residue of critique – the critique of commodity electronics characterised by features such as mass production, black boxes (or beige boxes), complexity, feature creep and vendor lock in. Commodity electronics are designed, manufactured and used in a way that overcodes and overengineers the strict scripts that structure its possible interactions with its environment. For instance, commodity electronics are supposed to be useful for all users independently of their level of expertise, and easily available through conventional market channels for the middle classes and upwards. Just to draw the most obvious consequences, both criteria require a shape and form that makes them fit easily within a cardboard box for transportation, and protected by cheap plastics against tampering. In other words, a blender or a hoover works and makes sense in a wide variety of social contexts.

Having said that, unfinished artefacts are not disfunctional and they do work. In fact they are more versatile because users can easily adapt them to changing situations, repurpose them according to their desires or redesign them according to their whims. However, dangling wires thrown around hackerspaces do not last as long as ones in plastic tubing plastered into walls of upper class residential buildings. Unfinished artefacts function by breaking down, which always serves as an opportunity to understand them better, to improve them somewhat or to come up with a better solution. They do require maintenance and repair but such maintenance and repair is more of a social occasion through which expertise is produced and shared in the context of a technologically prolific sociality than the subject of unexpected but necessary monetary transactions and alienated social encounters with the certified experts that one calls from the Yellow Pages. Almost all door system implementations above include a wide range of diagnostic functions and feedback mechanisms which tell the user about the state of the system. In contrast, take for instance the washing machines – they have many status indicators but none of them are explicitly diagnostic, and when they break down it is hard to know what happened, while self-repair is often thwarted by the lack of tools and replacement parts.

Unfinished artefacts can be finished as well. There are many reasons to close down unfinished artefacts, which mostly happens when they leave their native environment, typically to enter market circulation. In the case of door systems I have encountered two attempts to commercialise these small scale electronic artefacts. Visitors from an entrepreneurial society based in the local library saw the door system and access control implementation at TkkrLab, the hackerspace in Enschede, The Netherlands. They decided that they want to have something like that in their own work space and commissioned aps – a member of the hackerspace – to implement a "Lock-o-matic" for them, complete with space state that is displayed on their website, RFID authentication and automated door locks. Of course the ability of these new users to appropriate the door system technology will be much more curtailed than those of the hackerspace membership, and the technology itself will have to follow different design principles to work in a different social context targeting a different social group. Another attempt at finishing unfinished artefacts was recounted by my hosts at the Hack42 hackerspace in Arnhem, where the local secondary school ordered and installed a "Lanparty lock" in the computer room. The computers are isolated from each other on the network during classes so that students cannot play games with each other or hack each other's computers, but the last part of the class is reserved for exactly that. When the time comes, the informatics teacher takes out an enormous plastic key printed with a 3D printer at Hack42, and turns off the lock. From then on the computers see each other on the network and students can play together. The idea was inspired by the door system in the hackerspace and it was realised on their machines too. Despite the direct lineage, the Lanparty lock is immersed in very different social relations and supports very different social dynamics then the door system at any hackerspace. In particular, whereas the social function of the door system is to enable people to play with technology and to make their life easier, the Lanparty switch disables children from playing games together and thus makes their lives worse – the only person whose life and agency is enhanced is their superior, the informatics teacher.

Door systems themselves are a special species of unfinished artefacts. More than any other small scale electronic artefacts that are produced in the hackerspaces, they are entangled in the construction of unfinished architectures. Unfinished architectures are organisations that facilitate peer production practices at the same time as they are instantiating them. While hackerspaces have broken with their anarchist roots (as much as they ever had anarchist roots) in the hacklabs tradition, members still have a strong inclination to argue that the means should reflect the ends. Therefore, the production of hackable artefacts is best supported by a hackable organisation. Door systems mobilise objectified human agency to produce a more conductive environment for collaboration in a number of ways, including automating manual chores to posing technical challenged to members. But their most interesting role is in the coordination of the physical movements in the hackerspace which intervenes in the social dynamics of the membership. Hackers working on door systems are developing an artefact and shaping a community at the same time. While all pieces of technology change social relations in one way or another, in the case of the door systems both technology development and social change are explicitly acknowledged in practice by participants. Having established the concept of unfinished artefacts sufficiently, the next sections progressively take up the concept of unfinished architectures which shift the focus of the analysis from the artefacts in themselves to the organisations which give sense to them.

10.2 Door systems at the perimeter of hackerspaces

Since the previous section documented some instances of door systems, it is possible to look at them in the context of their everyday use. In a fundamental way, hackers build door systems because they consider it an integral part of hackerspaces, so when they build a hackerspace they also build a door system into it. Even though it is not part of any formal specification of how a hackerspace should look like, like the Hackerspace Design Patterns of Ohlig and Weiler (2007), for instance. But from a classic anthropological point of view, the small scale electronic artefacts that comprise door systems can be seen as ritualistic objects that mark the perimeter of the hackerspace in numerous ways that make them symbolically significant. Door systems are a necessary part of the passage from a rented club room to a veritable hackerspace (for groups), or from being an amateur engineer to a hardware hacker (for individuals). Taking the door system as a piece of material culture, I expand on the analysis about Technologia

Incognita to show the ways in which these small scale electronic artefact integrate into the life of the hackerspace.

10.2.1 Ur door system

Particular projects are structured by the patterns set out by unfinished architectures, but here we deal with one which also structures these patterns more than other projects. Door systems can be thought of as the project par excellence of hackerspaces. They also form the physical, virtual and logical opening of the hackerspace through which one enters this world. Door systems mark the passage from culture to subculture.

I have not been able to map the prehistory of door systems, which may be lost in obscurity. The Ur[137] door system must have been the entry gate at the main entrance of the c-base hackerspace in Berlin. Informants mention c-base as “the mother of all hackerspaces” or at least one of the first ones in Europe (1995), a honour that it shares with the Freaknet hacklab in Catania (1994), Sicily. The choice depends on the decision whether to construct the genealogy of hacklabs and hackerspaces as a single time line (Maxigas 2012a). According to legend, c-base is located in the remains of an ancient space ship lying beneath Berlin, a spaceship whose spire is the present day Fernsehturm TV tower on Alexanderplatz. The c-base association was founded to reconstruct the spaceship. At the entrance is a machine where visitors place their hands, upon which the machine displays marvellous sci-fi themed graphics and performs DNA analysis of the visitor to determine her life form. Then it arrives to the conclusion that the visitor is a human and greets her accordingly. The slogan of c-base is “Be future compatible!”

How to interpret this gesture? I identify three moments in its operation. Firstly, it encourages the visitor to experience herself in a wider context: as a member of the human species amongst all life forms in the universe, known and unknown. Secondly, technology itself is framed ambitiously in the context of the totality of all technologies, the possible and the not yet possible. Thirdly, the system connects humans and non-humans, that is the hackers, the aliens and technology, in a jocular and friendly manner, suggestive of a social space with universal aspirations. In effect, these moments add up to mimic the gesture of classic science fiction, but in a somewhat more performative manner, e.g. as *ritual* rather than fiction. The machine at the door therefore sets the tone for the activities which take place inside c-base.

As shown in Chapter 6, hackerspaces as a movement of proliferating hacker clubs only really started in the second part of the naughties (the 2000s). North European hackerspaces quickly reached a mass and consistency which shows that the social conditions were ripe. The door systems they use are very different

[137] I use *Ur* in the philological sense of being the first version of something that changed significantly after. The Urfaust of Goethe for instance is the first draft of the famous epic.

from the entry machine at c-base, yet perform similar gestures. The door system brings together *open participation*, which is supposed to be universal; *technological expertise*, which is supposed to be inventive; and *network sociality*, which connects the previous two. The next sections argue in more detail that the host of criteria already defined as the ethos of OSHW apply to door systems too.

10.2.2 Initiation

First of all, many members join hackerspaces to have access to knowledge and tools for building small scale electronic artefacts. Making a kit can be a first experience, and it is also a kind of initiation. Mitch Altman often advertises his workshops with simple kits using the slogan “learn to solder” and people leave such workshops with a badge/button sporting an LED and the legend “I learned to solder”, so that the badge/button is the medium to learn soldering but also the proof of honour that shows others that one has learned soldering. However, making and installing a door system is inevitably a collective project. Even if the bulk of the work is carried out by a single individual, it cannot be completed without coordination with others. Furthermore, even if the core system is built by one person, others will build on the top of that one extra functionalities that they are missing. Finally, since door systems just as much as anything built in a hackerspace eventually break, their maintenance have to be a shared responsibility especially because the whole membership and the visitors will have to rely on it.

Therefore a door system is often the first collective project of the hackerspace, and often the one which introduces some of the members to the experience of hardware hacking. Members do not necessarily know each other before they start a hackerspace together so it can be also their first experience in collaborating with the others in the group and discovering if and how they can actually work together. Making a door system is an important step internally in establishing the hackerspace as a physical infrastructure as well as a group of potential collaborators.

Externally, the first step is to register the new (or even planned) hackerspace on the hackerspaces.org aggregation site, which maintains a directory of hackerspaces. However, in order to show up in more sophisticated hackerspace statistics, and to make use of the multiple tools (like iPhone notification apps for instance) developed for door systems, the space would have to publish opening time statistics in a standard format defined by the SpaceAPI specifications. For all practical purposes, registering on the wiki and registering as a SpaceAPI endpoint marks the formal entry to the hackerspace into the scene. There are many other ways in which hackerspaces are networked, from online spaces such as Internet Relay Chat and mailing lists to offline spaces like hacker conventions, exchange programmes and so on and so forth. However, people participate in these individually as one member of the hackerspace, so they do not bear directly on establishing the hackerspace as a hackerspace.

These technical steps are even more important since there is no formal agreement or bureaucratic procedure for becoming part of the hackerspaces scene. Programmers would call this kind of identification “duck typing”: if it looks like a duck, if it acts like a duck, then it *is* a duck. A new hackerspace would simply have to gain enough family resemblance to other reputable hackerspaces in order to be recognised one.

Being part of the scene means a steady stream of visitors, workshop offers, donations of equipment and the like. Potential new members moving into another city shall find the hackerspace more easily and perhaps join. All these people look at the website and then the space state on the website to know if the hackerspace is open. Even if they eventually visit the space on a “social night” traditionally held on Tuesday evenings which is like an regular opening time for many hackerspaces, statistics about how often the space is open are a good proxy for measuring the activity of the particular hackerspace.

Finally, the door system has to exhibit the design principles that pertain to unfinished artefacts and peer production projects in general, so that they set the tone for further creations. Simplicity, modularity, granularity, loose coupling and transparency in the technical composition as well as the documentation have already been mentioned and demonstrated in specific cases. While these sound straightforward, they go against the values ingrained in pupils of mainstream engineering education as much as many of the tools available on the market to do the job.

10.2.3 Sustenance

After the establishment of a hackerspace, the door system comes to mark the symbolic boundary of the hackerspace. It is the first and the last thing that visitors to the hackerspace encounter, and it is often the infrastructure that prevents or enables them to enter the architectural space of the hacker club. Members usually give a tour of the hackerspace to visitors who have not been there before, including the door system and its operation. In line with the hands-on attitude of hackers it is customary to explain not only what the door system does, but also how it works and what are the plans – if any – to improve it, or how it fails in various cases. Upon joining the hackerspace, new members are introduced to their rights and responsibilities, like the protocol to follow when opening and closing the space. These include a list of tasks like the one cited in section 10.1.5, often posted next to the door, as well as pushing the space state button.

Many people associated with the hackerspace in one way or another mostly participate through the chat channel of the hackerspace only. Even though they seldom find the time to attend the hackerspace itself, they are part of the online conversation and help out when needed. The space itself is logged in to the channel as a “bot”, a program that masquerades as a person announcing the opening and the closing of the space and usually taking on other automated tasks like citing the title of the relevant web page when any URL is pasted by a participant of the channel. So from a remote perspective the hackerspace is impersonated by a bot (in Foulab, Montréal, Canada the bot is called foubot, for instance) hooked up to the door system.

It is easy to see that in most of its implementations the door system switch stands out of its environment somewhat. The elaboration and theatrical presentation that goes into a door system marks it as symbolically important. On a fundamental level, the door system turns on the hackerspace. Since object are not worth much without subjects, in the same way that the sound of a falling tree have to be heard to have any social existence, the hackerspace does not exist as a hackerspace until the door system sets the space state to open. As an open space for socialising and collaborating around technology, the hackerspace does not fulfil its mission until some hackers are in place. Therefore switching the space state is a magic moment that effectively brings the hackerspace into existence.

10.2.4 Social shaping

The primary use case for the door system is basically to let members know when somebody is in the space. Since many members would not go in just to sit alone, they prefer to wait for others to turn up and then they are themselves more motivated to go to the hackerspace. In fact it is comparatively hard to work in a hackerspace alone, because when others are there it is easier to ask for their advice, get access to various resources or simply learn where things are. It is common to see on hackerspace chat channels requests for the whereabouts of this or that tool, or plea for help to people who are known for their expertise in a specific area. Such queries may receive answers in a few minutes, in a few hours, or not at all. Therefore, it is better to ask people who are present in the building, if any.

On the other hand, some committed members do exactly the opposite. As they feel important to “keep the space open”, they would go to the hackerspace when they have time and they see that nobody is there, just to enable and inspire other people to come. In this way the door system nudges members to participate in the hackerspace. As mentioned in Chapter 6, many hackerspaces have members who contribute less through their technical expertise but spend much time in the space, which is valued as a worthwhile activity in itself called “providing uptime”. There is a whole ecology of humans, machines and machine parts moving in and out of the hackerspace, and it is sometimes very useful if somebody can just be there and follow the chat channel at the same time.

As much as the hackerspace is a collective medium where the reputational capital of members’ projects adds up to the reputation of the hackerspace itself, the chat channel is named after the hackerspace and it serves a collective voice for the hackerspace. In case somebody wants something from the hackerspace, it is best to ask there because there will be more members present than in the space itself. Even though sometimes members point out specific persons who are the ones to talk to about specific topics, most often it does not really matter who is answering a given query or question as long as they are familiar with the situation. Therefore, in case one does know each member personally, the various nicknames of people and bots on the channel does not have much meaning: it is as if the hackerspace itself would answer. Notably, this is a rather unique feature of Internet Relay Chat compared to other real time chat services like Facebook or Google chat, or Twitter for that matter.

IRC is an online coordination tool for the hackerspace where many signals meet, generated by humans and non-humans alike. Door systems are tangible interfaces to trace the physical movement of members and the activity of the hackerspace. Signals from the door system feed into IRC and other online spaces. Sometimes signals from the online space of IRC feed into small scale electronic artefacts in the hackerspace too. Audible or visible can be generated through bells and whistles in several hackerspaces. For instance in RevSpace (the hackerspace in The Hague, The Netherlands) there is a rotating emergency light which can be activated to call the attention of members to events like an imminent pizza delivery. Additionally, there is a huge LED scrolling display in the back wall of the main room which members use to send messages from the IRC channel to the others sitting in the hackerspace who may not follow the disembodied conversation there. In hackerspaces where there is a vocoder (computer voice audible in the whole hackerspace) installed like HSBXL or the London Hackspace it is also possible to synthesise voice messages to the people in the space. These Augmented Reality devices blend online and offline spaces to coordinate between hackers in and out of the hackerspace itself. These infrastructures contribute to the social relations unique to hackerspaces which are conductive for peer production practices. Building, using and maintaining these infrastructures makes what I propose to call an unfinished architecture, initiating members to the form of life that is specific to the hackerspace and integrating the hackerspace itself to the particular techno-social network that constitutes the scene.

10.2.5 Conclusion about the ritualistic roles of door systems

The actual functionality of door systems is primarily to signal space state to people who are away, and secondarily to share meta-information about the space using various tangible and logical sensors, or to control events like access authorisation through tangible or logical actuators. I have argued that beyond these actual functionality door systems perform a number of social functions in the ecology of hackerspaces. If hackerspaces are defined as a concrete scene tied together by a techno-social network, door systems play a central role internally to the hackerspace and externally in relation to other hackerspaces in bringing a new hackerspace and its members into the cultures and infrastructures of the hackerspaces scene. One, in concrete terms becoming a hackerspace means appearing on the various databases of hackerspaces, one of the most significant of which is the SpaceAPI which integrated space state informations. Two, a hackerspace as a hacker club open to the general public is a hackerspace only as long as space state is reported, since regular opening times are scarce, while both members and visitors are eager to know what time they can access the material infrastructure of the space itself, and meet like-minded people on the premises. Three, founders of the space are normally not already in cooperating terms with each other and they are not necessarily familiar with the tricks and trades of hardware hacking, so that building a door system as a first collaborative creation of an unfinished artefact establishes them as both hardware hackers and as a collective of collaborators that make up a hackerspace. Four, the door system as one of the most visible and ubiquitous projects in the hackerspace embodies, performs and displays the engineering principles behind the construction, use and maintenance of unfinished artefacts, serving as a blueprint for further future creations.

There are other points that are more important once the initial roles of the door system are fulfilled. Five, door systems are instrumental for the gatekeeping practices of the community around concrete hackerspaces, whereby principles of gatekeeping are inscribed into the technical functionality of the door system which continues to embody, perform and enforce them. Six, door systems grow in time to encompass the whole space through tangible and logical networks, bridging online and offline spaces as mediators of transactions. Seven, developed door systems transform human tasks to tasks for non-humans without necessarily stripping away human agency by closing their architecture, as it often happens when automation sets on. Eight, door systems become on of the default options for hacking on something because when attendant hackers do not find a concrete project, they look around hack the building itself. Nine, switching on a door system toggles the space state, which can be seen as a metaphysical ritual whereby the hackerspace comes to actual existence as an active space of collaboration in technological creativity – a function that is emphasises in the theatrical presentation of tangible interfaces to door systems such as huge buttons. These points identify the ways in which door systems integrate into the everyday social life of the hackerspace internally, and the ways in which a more or less consistent engineering culture as well as a more or less consistent material infrastructure is cultivated in order to sustain the scene. The following sections pick up one or the other points here to develop them further, or depart from these established notions to build up arguments.

10.3 Extensions of door systems

While the first empirical section concentrated on local variations between door systems and started from the implementation of their basic functionality, this second empirical sections collects common extensions of door systems to further understand how they come to encompass the hackerspace and structure social interactions. These also allow the analysis to slowly shift from unfinished artefacts to unfinished architectures: environments conductive of peer production practices. Finally, as these initiatives grow from small scale electronic artefacts to fully fledged large scale material infrastructures, they also testify to the notion that despite the great variety in their implementations, unfinished artefacts are nonetheless functional, and can be even compatible with each other.

10.3.1 Space statistics beyond space state: Moar sensors

Hackerspaces routinely expand the range of statistics gathered and published – sometimes with good reason, and sometimes simply because it looks like an interesting experiment (e.g. "because they can"). Sometimes building a sensor system and gathering data is a fulfilling end in itself, as part of socialising with others, learning and discovering new things. Some other times it is a way for hackers to come up with a technical answer to a perceived problem.

An instance of space sensors in the form of temperature sensors covering a great number of rooms at Hack42 have already been cited in Section 10.1.4. Another case in point is the "water percent" counter set up in H.A.C.K., the hackerspace in Budapest, Hungary. The hackerspace is located in a basement under a residential block. One day water was leaking from the ceiling due to a mishap, resulting in the flooding of the hackerspace. Following claims that the problem have been dealt with there was still drops from above. A bucket was set up to catch the water and an indicator built using simple electronic components that reported the percent of water in the bucket on the chat channel of the hackerspace. This allowed members to check on the unfolding of the potential disaster even while they were away from the lab. Such a haphazard contraption is only possible where the necessary expertise and material infrastructures are readily available. These enable members to adopt to unforeseen situations in experimental ways on the spot.

Sensors may give members a way to get a closer grasp on their hackerspace which develops forms of ownership and belonging. The ability to get data from their environment can lend a sense of agency to being a member of the hackerspace, even though privately identifiable data gathered in the lab can have the opposite effect. As already mentioned before, the amount of data published on a hackerspace is an indirect measure of its activity, but may also reveal interesting patterns about the use of the space. These aspects resonate with citizen science initiatives like the open source Geiger counter developed by a Japanese hackerspace documented by Kera, Rod, and Peterova (2013) or the Smart Citizen Kit developed mainly at Fab Lab Barcelona (Lanzeni 2015).

Since space state is usually the first type of statistics to be produced from the hackerspace and the door system is the first piece of infrastructure put in place to telecommunicate what happens in the lab, additional sensors often get connected and statistics delivered through the the door system implementation. London

Hackspace uses, as this excerpt from the chat channel shows:[138]

```
20:37 <ham_sandwich> ?laser
20:37 <robonaut> Laser cutter in service and currently in use
```

ham_sandwich is a human user who asks the bot on the channel in charge of setting the space state for the status of the laser cutter. Questions to bots conventionally start with a special character like the question mark so that they are easily distinguished from normal conversation. The bot answers that the laser cutter is not broken at the moment but somebody is using it right now. Since laser cutting can be a lengthy process this could indicate that it is not worth going around to the hackerspace for doing some laser cutting, or a member already in the space could use the same function to know when to go over to the next room to use the machine too. While convenience functions such as these address practical problems they also serve as toys for learning and experimentation or simply interacting with the space itself.

10.3.2 SpaceAPI: Large technological infrastructures

The SpaceAPI initiative was started by members of RevSpace (The Hague, The Netherlands) and Bitlair (Amersfoort, The Netherlands) as a way to increase interaction between hackerspaces. Advertised as a “decentralised information system for hackerspaces”[139] it effectively establishes a level of abstraction where the radically different door system implementations can be reached through a unified interface. The idea is for every hackerspace to publish a file on their website containing certain required and optional fields set in the format defined by the SpaceAPI specification. Concretely, the water percent counter from H.A.C.K. that allows hackers to keep a remote eye on the leak in their ceiling is published by the endpoint of that hackerspace in the following stanza:[140]

```
"sensors": {
    "humidity": [
        {
            "location": "Bucket",
            "unit": "%",
            "value": 0
        }
    ],
```

It is easy to see that the format (a light-weight contained format JSON) is intended to be readable by both humans and machines, and structured so

[138] Chat log, irc://irc.freenode.net/london-hack-space, 2015-05-18.

[139] https://spaceapi.net/

[140] Taken from http://vsza.hu/hacksense/spaceapi_status.json“, visited 2015-05-20.

that the list of “sensors” includes a “humidity” sensor whose location is in the “Bucket”, containing zero percent of (presumably) water. A sensors section can contain an arbitrary number of sensors. Similarly, space state is represented like this:

```
"state": {
    "lastchange": 1431956220,
    "open": true
},
```

The date format is in Unix time, which is the number of seconds since January 1, 1970, midnight UTC (Coordinated Universal Time), and the “open” is a Boolean (true of false) variable. The current version (0.13) of the SpaceAPI specification defines 8 required and 9 optional fields. For instance space state is mandatory but sensors can be left out. SpaceAPI usage is invariably tied to door system implementation because the space state is the only required field which is dynamic – other required fields merely provide the contact details of the hackerspace like geographical location, physical address, website address and similar. Therefore, hackerspaces have to implement the basic functionality of door systems – reporting space state – if they want to be part of the SpaceAPI network.

In this way the SpaceAPI introduces a path dependency that gives further motivation for implementing door systems. The social function of the SpaceAPI is to provide a coherent way to learn about and interact with hackerspaces despite their differences in general and the wide range of door system implementations in particular. Once again, since there are no formal or bureaucratic requirements for becoming a hackerspace, integrating into the technical infrastructures of hackerspaces is an important indicator for being perceived by practitioners as an authentic hackerspace. These observations support the argument that the door system is a gateway for entering the hackerspace scene, mainly because the contact details are only disseminated through these channels once there space state is reported in quasi real time.

During my field work in North European hackerspaces I encountered a few hackerspaces that do not implement SpaceAPI but implement door systems, and none that had no door system at all. At the time of writing 147 hackerspaces provide valid SpaceAPI endpoints, which is a small number compared to the 1153 active hackerspaces registered on hackerspaces.org. While SpaceAPI-enabled hackerspaces can be found on all six continents, it appears that they are concentrated in North Europe, therefore rather specific of my field. The decentralised but not distributed because each hackerspace uses their own hardware and software infrastructure to gather the data and publish their information – the central SpaceAPI website merely provides the specification and publishes a list of registered endpoints. In turn, the development process of the specification is open since anybody with an interest in the project can make feature requests through the Github software repository service, talk to the developers and potentially become one through joining the dedicated chat channel, or subscribe to the mailing list for a more deep involvement.

Finally, it is evident that OSHW is most effective in creating social interactions when it is integrated through FLOSS components. Therefore, OSHW cannot be studied without attention to the software aspects of the infrastructures people build using these tools. Unfinished artefacts radicalise the OSHW ethos by encoding values of alternative engineering cultures in the technical composition of small scale electronic artefacts, not just securing the rights of users through legal means and the availability of documentation as a knowledge commons. Social relations which are constructed through systems like the SpaceAPI come together into unfinished architectures which in turn provide an environment where unfinished artefacts can work well despite their brittleness and variations in terms of functionality.

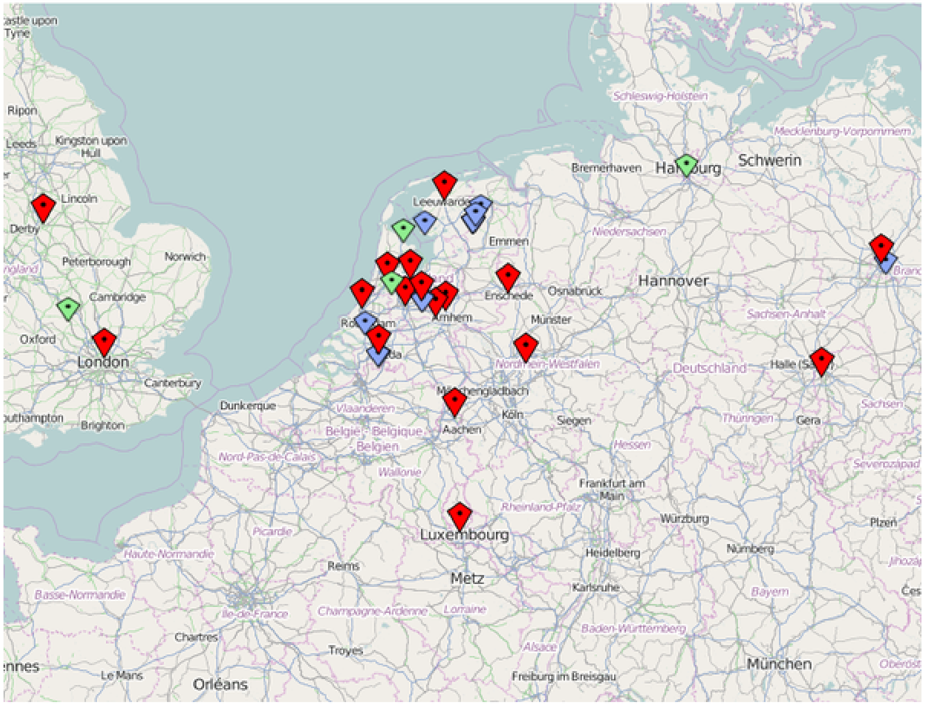


Figure 28: Map of all SpaceFED access points. Legend: red markers are SpaceFED in hackerspaces; blue markers are SpaceFED in other hackerspace-related spaces. One green marker that stands for a Hack in the Box meeting in Kuala Lumpur, Malaysia is not shown. Licence: Copyright. Author: SpaceFED contributors. Source: https://spacefed.net/wiki/index.php/Who/Spacenet/APs

10.3.3 SpaceFED: Distributed authentication

The SpaceAPI specification includes a SpaceFED field whose documentation explains the latter project succinctly:

> A flag indicating if the hackerspace uses SpaceFED, a federated login scheme so that visiting hackers can use the space WiFi with their home space credentials (Space API authors 2015).

As an eminent unfinished artefact, the coherent (but still decentralised) information space provided by the SpaceAPI allows hackers to build higher systems of abstraction on top of it, using the same design principles. Hackerspaces in SpaceFED have wireless routers (access points) that require a username and password to connect. These credentials are handled by an authentication server that is also reachable from the global Internet. In case a user account is not found locally, SpaceFED looks at the name of the account which contains a pointer to the authentication server of another hackerspace. Much like with emails, the username "foo@bar.org" will be authenticated with the server reachable under the address bar.org. The obvious benefit is that each hackerspace can manage their own membership database autonomously, while members can use the wireless Internet access at any location where SpaceFED is available. The technical term for such an authentication mechanism is federation.

The SpaceFED system closely resembles another federated authentication system called `eduroam`, which enables university students to get Internet access not only at their home institution but in any place in the `eduroam` network. As mentioned before, SpaceFED enabled networks are often installed by hackers outside of their hackerspaces too, usually at hacker conventions where network connectivity is in short supply and unstable because so many people use it in so many ways. SpaceFED provides its users a more stable and secure channel to the Internet and also to the local network of the hackerspace. Therefore, SpaceFED is also a way for bringing the hackerspace with you to the field. This is all the more true because in the same way that students who use `eduroam` get access to additional services like repositories of academic articles which are not available on the open Internet, hackerspace participants can access their local services such as media servers or the door system itself through SpaceFED. Technically, this is done through a VPN (Virtual Private Network) which bridges the client computer with the local network without disabling normal Internet connectivity.

As a clone of `eduroam`, the SpaceFED initiative represents the hackerspaces' challenge to the modern institution of higher education, or more precisely, a rearticulation of the aspects hackers find valuable in it. However, as an unfinished architecture, in contrast with `eduroam` it allows any hackerspace or other interested community to implement its requirements and join the network without any bureaucratic hindrance, request for authorisation or fee payment. After joining the network, there is literally nothing to prevent new members to implement new services and start to provide them to SpaceFED users. Moreover, SpaceFED is documented on a public wiki and free of charge community support is provided through a chat channel and a mailing list. These properties follow the rights of users enshrined in the GPL, i.e. to be able to understand, modify and reproduce technologies.

Moreover, as another aspect of the techno-social construction of the hacker scene, SpaceFED is an exemplary gatekeeping mechanism. In fact its algorithm closely mirrors and therefore reinforces the social boundaries and gatekeeping practices within and without the hackerspace communities. These social boundaries can be reconstructed as follows. Hackers join a concrete hackerspace as a member and receive benefits from the unfinished architectures (material infrastructures as well as social relations) provided by that hackerspace. At the same time, as members of one hackerspace they become a more welcome visitor in other hackerspaces where it is more easy for them to gain the necessary trust for partaking of the same benefits as they enjoy in their home base. When hackerspace members come together to join forces at a hacker convention, being part of the hackerspaces scene enables members to engage with each other more easily, which makes for a more enjoyable experience. The algorithms that operate authentication mechanisms in SpaceFED can be seen as a technical encoding of these boundaries, but of course at the same time they also reinforce them. A trivial example is going to a hackerspace one has never visited before, being surrounded by strangers but being able to access the Internet right away. When people in such "foreign" hackerspace learn that the visitor is a member of another hackerspace with (a.) a door system, (b.) a SpaceAPI integration, and (c.) a SpaceFED authentication, they feel that they belong to the same scene as the visitor – and group solidarity ensues. Addressing a central concern of Science and Technology Studies, the structural symmetry between social boundaries and gatekeeping practices within and without the hackerspace communities highlights how the encoding of shared culture into technological artefacts works within the hacker scene.

SpaceAPI has been implemented by 20 hackerspaces and used in about a dozen events since its inception in 2012. Geographically, it has also been incepted in The Netherlands and seem to emanate from their, not even covering Eastern Europe (Figure 27). As yet another layer of abstraction, now built on top of the SpaceAPI infrastructure, SpaceFED requires a more sophisticated technical infrastructure to be in place at a hackerspace that would join – and at the same time, more trust built between hackerspaces, because it is about automatically giving access to local resources to members of other hackerspaces. Technical workarounds aside, in case the social relations between hackerspaces are too weak for a particular hackerspace to trust all the other arbitrary number of hackerspaces that participate in SpaceFED just because they are hackerspaces, it is probably not a good idea to implement SpaceFED in that hackerspace. Once again, unfinished artefacts are working very well in the context of unfinished architectures, but as long as they are taken out of context, doubts about arise.

Compare the case to eduroam. One could argue that unfinished artefacts and unfinished architectures are useless conceptualisations because no device can work and no device can be understood outside of its social context – and there would be merit in such an argument. Perhaps these terms only remind scholars of self-evident truths that should not be forgotten, illustrating theses that are already told. In order to cite concrete empirical proof for such an argument, one could argue that eduroam is built on a very similar architecture and it is providing very similar services. However, there are three crucial differences. Once, a legal difference is that eduroam is backed up with a system of contracts and a central administration that stabilised the network. Even if its technical architecture is comparable to SpaceFED, it is effectively neutralised by a hierarchical management structure which closely resembles the hierarchical social structure of the universities that use it. Twice, the technical architectures are in fact different because even if a university student can get Internet access as well as the local services provided by her university from any eduroam member network, connecting to eduroam can never unlock services of any third eduroam university. On the contrary, connecting to SpaceFED can and does provide access to third party services that are not operated by the member's own hackerspace nor the other hackerspace that the member would be visiting – so the guest and host network operators retain control over the range of services they provide. Thrice, one has to see that while all machines depend on their social context, most are secured and stabilised from interventions by their users exactly to keep them working. In contrast, unfinished artefacts depend exactly on the social relations between their users for their functionality. This point is pushed further in the next and last subsection.

10.3.4 Access control: Solidarity of clubs

While SpaceFED is the youngest initiative of systems integration that came out of door systems – which probably accounts for its lower adoption rate and narrower geographical reach as of the time of writing (together with its higher level of abstraction as already argued) – there are current plans on how to take the idea further. During the same period when the SpaceAPI line of initiatives has developed within the hackerspaces scene, an unrelated trend was the installation of electronic locks coupled with RFID authentication devices at the doors of hackerspaces. These initially simple solutions grew incrementally to rely on aggregated member databases kept by the respective hackerspace. Since SpaceFED already implemented credentials sharing, cross-hackerspace compatibility of electronic keys became a viable option. Linking the "network security" aspects of SpaceFED to the "physical security" aspects of electronic access control mechanisms would allow a key-holding members of one hackerspace to be able to unlock the doors of other hackerspaces. Especially because the technical solution is fairly trivial to implement in hackerspaces which have all the previously mentioned systems in pace, the case highlights the social relations at stake in pushing the limits of unfinished architectures through unfinished artefacts.

The plan is not as far-fetched as it sounds, however, since there are historical precedents that demonstrate inter-club access based on solidarity to be implemented in practice. It has already been mentioned that some high-profile members of Noisebridge (the hackerspace in San Francisco, CA, USA) were distributing the actual tangible keys to their hackerspace to the audience after presentations, following the phrase that "Our doors are always open to you". It has also been mentioned (in Section 7.6) that long before Noisebridge, members of Gentlemen's Clubs who were visiting foreign cities were entitled to the same services at reciprocal clubs that they would enjoy at home, including lodging, dinners and other amenities. Reciprocal agreements served as social pedigrees which established a line of trust, ensuring that the visiting member is a respectable gentleman of the breed that meets the expectations of the other club. There is an element of the same in the algorithm that the federated authentication protocol asks local servers for credentials first and query remote servers second in order to determine that the user is a known hacker of good standing, but also in the informal vetting processes and gatekeeping practices through which membership in one hackerspace provides rapport for visitors to another.

The arc that door systems development drew out highlights the co-construction of unfinished artefacts and architectures, since the very process of building the unfinished artefacts together was the most important experience through which the necessary trust for further steps towards the radicalisation of the unfinished architectures have been established. Finally, the cycle of technologies also came full circle: setting off from the OSHW implementations of door systems through using FLOSS components to unify unique instances across hackerspaces ended up at the OSHW operating the very doors once again.

10.4 Distributed peer production practices

As the r0ket study served to establish the fundamental difference between the peer production of hardware from the peer production of software that hackers need initial investment and fixed capital to manufacture devices in sensible numbers to make them actually useful for communities, and to show how this effects the technical composition of small scale electronic artefacts, the door system study points to an alternative scenario of distributed manufacturing that is also deeply different from mainstream narratives of software production. The main moral of the door system study is also rather straightforward: while software can be copied at no cost from one place to another (the very property that early peer production theorists thought makes it fit for peer production), each copy of hardware has to be reproduced from scratch. One option, the introducing industrial factory production into the equation has been explored by the r0ket team. Another setup is explored here where reach hackerspace produces their own door system implementation that responds to the local context, ranging from the personal interests of participants to the political economy of the respective nation. Therefore, rather than free software which is often copied as a whole and developed incrementally based on a local copy of its cutting edge version, door systems are typically implemented from scratch by people who are more or less familiar with a range of similar solutions. This has a number of consequences, but first an aside on free software.

Even if I claim that OSHW is reproduced from scratch more often than FLOSS, this does not mean that comparable examples cannot be found in the development practices of free software. In fact, it does not even mean that free software is typically developed as it is described in most social scientific accounts. There is a whole mezzo-level of free software development whose routine day-to-day practices are not recorded simply because projects are not famous enough or not numerous enough to turn up in qualitative studies – which typically look at only the outstanding examples that show up on the researchers' radar – or qualitative studies – that take whole domains as their unit of analysis and routinely fail to recognise and theorise local logics. The ecosystem of static blog generators – a simple sort of blog software that generates HTML files – is a great example. Static blog generators learn and copy from each other, and while the fundamental idea remains the same, it is implemented in a thousand different ways. Even though there are more and less popular static blog generators, the distribution of users is such that a great number of these software applications have their own user based and sometimes even communities. In fact, the complexity of the problem is such that once a practising developer grasps the basic idea, she often decided to roll her own implementation instead of choosing from the wide range of already available solutions. However, static blog generators are out of the scope of this investigations, they are only to point out that a similar dynamics to door systems can and do exists in software, even if not necessarily documented.

Returning to the consequences of distributed practices of peer production to OSHW and by extension to unfinished artefacts, I claim that the door systems show a widespread model of working on smaller scale projects which have not been widely investigated in scholarly literature. Distributed manufacturing of OSHW have been proposed theoretically by Gershenfeld (2005) and Anderson (2014) for capitalists, as well as by Rigi (2012) and Dafermos (2014) for anti-capitalists. It is used in practice by high profile projects like RepRap (3D printer), Wikispeed (car), and various Arduino clones (microcontrollers). Distributed manufacturing differs in several important respects from the model of distributed practices of peer production of unfinished artefacts which is epitomised by door systems.

In distributed manufacturing the same design is reproduced in many places, so that local manufacturing centres like hackerspaces (Rigi 2012; Dafermos 2014) or Fab Labs (Gershenfeld 2005; Anderson 2014) replace factories. The central claim of the four authors mentioned above is that distributed manufacturing can displace industrial aspects of capitalism by merging them with cognitive capitalist knowledge-based production practices. All of them agree that such as scheme would work in both rich and poor countries – in the centre and in the periphery of the division of labour that characterises global capitalism. Capitalists present these claims in the language of disruptive innovation for liberal democracies and in the language of developmental discourse for failed states. Anti-capitalists frame it as a revolutionary social process that need to enroll the local state and local capital in its project (Dafermos 2014) or confront them in a revolutionary show-down (Rigi 2012). However, they do not look at the social basis of peer production practices, only present technological determinist arguments for its spontaneous emergence thanks to the inevitable development of fixed capital and its eventual victory thanks to its superior efficiency. While Rigi, for one, recognises the contradictions in capitalism that peer production practices could possibly articulate, and hence he can conceptualise a revolution with social conflict on a historical scale, he fails to account for the social process which produces peer production practices themselves.

I have tried to trace in the preceding chapters (Chapters 6, 7 and 8) the formation of the hacker scene in order to place these developments in the context of social history. Hackers as a particular social group – some of the most privileged workers at jobs which are ideologically overpaid because of the role of ICTs in contemporary capitalism and somewhat hard to penetrate for management because being knowledge intensive thus granting enough free time for developing a semi-independent culture – have struggled for decades in order to be in a position to found hackerspaces which provide conductive environments for the peer production of unfinished artefacts. The process of institutionalisation created organisations which can mediate between the interests of the state and capital and the interests of hackers, as well as translate cultural meaning assigned to forms of organising labour such as peer production, or technological practices such as reverse engineering. Of course such a process is not without conflict and failures: in many cases it cannot even defend its constituency from direct repression by the state and capital. Moreover, institutionalisation does not make the hacker scene immune to recuperation, especially once it has established itself as somewhat of a social force. Indeed, the quasi-institutions formed by hackers can serve as an interface for recuperation by the state and capital depending on how strongly the relative autonomy of the scene is articulated in the given context. However, the historical process of social formation can hardly be replaced by the issue of a franchise licence such as in the case of Fab Labs. The difference between the Hackerspace Design Patterns of Ohlig and Weiler (2007) and the Fab Lab Charter is that the former gave coherent form to organisational practices deeply rooted in an alternative engineering culture with its own history and infrastructure, while the latter served as a manifesto to kick-start such alternative practices of manufacturing and knowledge production. This is evident in that the former concentrates on middle range theory to resolve common organisational problems without any mention of machines, while the latter goes at lengths to specify the exact machinery to be acquired.

Technological determinist arguments aside, there *are* concrete material conditions which social formation exploits in order to move forward. The argument of von Hippel (2005a) that the cheap availability of basic electronic components in the consumer market played a crucial role in the rise of user-centred innovation is convincing. However, these material conditions are necessary but not sufficient for explaining or even interpreting what came after. Hackers struggled to find ways of encoding values in small scale electronic artefacts that are different from mainstream engineering practices and design principles. They gave new meaning to and found new technical compositions for the electronic parts available on the market. These values are meaningful in the context of the hacker tradition, and not initially supported by the products offered on the market – something that obviously changes with increased demand for certain kinds of products, or products with certain properties that make them more easily hackable.

In turn, the historical formation of a social group is what makes it possible for peer production practices to be actually distributed, not merely the manufacturing capacity which can be established through investment in fixed capital. Hackerspaces provide the material infrastructures as well as the cultural milieu for the proliferation of alternative engineering practices that seamlessly combine education, research and manufacturing. In the case of the door systems this means that there is enough local knowledge, materials and last but not least human free time to implement the basic idea from scratch, while taking into account the lessons learned from the state of the art of previous implementations. Distributed practices of peer production of small scale electronic artefacts allow for adopting the idea to the local conditions across multiple scales from individual interests to geopolitical positions.

The arguments of transaction costs theory rest on the universal properties of certain ICTs that make them more efficient for organising production in market-like conditions which are also deemed universal.[141] That is how proponents of transaction cost theory can propose a universal solution without geographical and historical specificity, one that can speak to theorists of all political orientation. This might actually work for the distributed manufacturing of OSHW, but not for distributed practices of peer production for unfinished artefacts. What I argue is that peer production practices are enabled by the formation of social groups, organisational forms and material infrastructures which are the result of a specific historical process. This opens two lines of critique towards proposals of distributed manufacturing.

On the one hand, there is a range of proposals to enhance various institutions with shared machine shops that span the whole range of the modern institutional grid. Projects to establish shared machines shops at the sites of capital like major

[141] The critique of transaction costs theory is worked out in more detail in the theoretical framework in section 3.3.1.4.

corporations and sites of knowledge production like universities to boost their innovation potential. Projects to establish shared machine shops at libraries to further the project of increasing public participation in science and technology. Projects to establish shared machine shops at the sites that are seen in need "development" like ghettos, depopulated countrysides, war zones or failed states. The common thread running through these ponderous proposals and pilot projects is that they concentrate on distributing the product and not the process. Ironically, a call to turn churches into hackerspaces for the co-construction of theology by a United Methodist clergy-person makes the most sense (Smith 2013).

If peer production practices are the result of a social formation, then it is not shared machine shops that people need but a proliferation of social relations that enable specific social groups to develop alternative engineering practices and establish a relative autonomy in relation to the state and capital. Once alternative conceptions of technology are rooted in a specific milieu, these communities of practice can exploit the potential of ICTs which are a necessary but not sufficient condition for unfinished architectures where unfinished artefacts can be produced. Otherwise, planting shared machine workshops merely distributes the manufacturing capacity – as in distributed manufacturing – but does not proliferate the social relations that make unfinished artefacts make sense in the local context. Practically, even if OSHW designs can be reproduced locally, local users have to make sense of them in the context of their everyday lives and reimplement the ideas in a way that makes sense in respect to the material conditions on the ground. Door systems are the backbone of hackerspaces but would falter in many other contexts. Similarly, r0kets tie together the people pertaining to a specific engineering culture but users belonging to other social groups do not see themselves in it and consequently they would have no idea what to do with an electronic conference badge.

These conclusions open the way for a second line of criticism. As the case study of the r0ket has shown, the manufacturing capacity manifested by a single hackerspace is scarcely enough for the production of small scale electronic artefacts that can serve individual members of the local community in a meaningful way. Furthermore, the door system example showed that at least in the specific socio-cultural context of the hackerspaces, reproducing the same thing over and over does not even make sense for people as long as they are empowered by an alternative conception of technology, engineering expertise, structures of participation and material infrastructures – in short, what I call unfinished architectures. Bereft of factory automation, as the r0ket makers already recognised through their own experience, the only way to make many copies of something in a hackerspaces using the signature tools of the "new industrial revolution" is through crafts-work reminiscent of manual piece work in sweat-shop settings. Even if using a 3D printer or a laser cutter is more glamorous than operating a lathe machine, the hackerspace remains a medium for making prototypes and rather unique unfinished artefacts. Hackerspaces falter when they are expected to perform the work of a research and development department in a big corporation or as factory – as they are expected to do by theorists of distributed manufacturing such as Rigi (2013) or Dafermos (2014) both because mass production does not fit their institutional cultures and because the fixed capital they control have not been designed with mass production in mind.

As a corollary, note that the net effect of these two mistakes closely reproduces the paradigmatic shift in the dynamics of the global distribution of labour that characterises late capitalism. Knowledge production which constitutes an increased source of wealth remains in the centre while manufacturing capacity is pushed out to the periphery where it may or may not improve local conditions. There is no easy solution to propose in terms of ICTs for development, however. The hard truth that analysts and activists have to confront is that at least according to the case studies explored here, peer production is tied to the ascendancy of a particular social group of privileged workers. Fortunately, this does not necessarily mean that unfinished architectures conductive to the production of unfinished artefacts are not possible to implement outside of core countries in Europe and North America. Lindtner (2014) makes a convincing case for the existence of production practices that are deeply embedded and productive of social relations within the innovation ecosystem in Shenzen, China. Crucially, she argues the case in the context of local developments in political economy that are understood in the context of the division of labour of global capitalism. She gives a geographically and historically specific account of the rise of peer production practices in the Shenzen area of China which is tied to the formation of a particular social group.

In summary, distributed practices of peer production of unfinished artefacts in the case of door systems emphasises the primacy of social relations, while distributed manufacturing of OSHW concentrates on the agency of fixed capital to achieve similar aims. The unfinished artefact is proposed as a more rounded conceptualisation of OSHW that privileges the social relations that are co-produced through peer production practices, and unfinished architectures to refer to organisations conductive to such practices. Existing theories of peer production have to be developed in a way that can understand the phenomena at hand in the context of social history as a project of a particular social group taken in its geographical specificity.

10.5 Translating human and non-human scripts

Since Bruno Latour wrote at least two articles (1988b; 1992) that engage explicitly with door systems in order to work out a (now somewhat dated) language of the participation of machines in human life, it is tempting to offer a closer commentary on his conceptualisations. Both are built around the case study of a broken groom (automatic door closing contraption) which was replaced by a hand-written note exclaiming `"The Groom Is On Strike, For God's Sake, Keep The Door Closed"`. On the one hand, the cases are close enough to the door systems discussed here to warrant a cross-examination that can show the particularity of the social contexts. In this respect it is interesting that Latour rewrites the same case study twice, once placing it ironically at the religious Walla Walla University in Washington state, USA, and the second time to the La Halle aux Cuirs in the La Villete, Paris. Despite allusions to the spatial context of both doors, neither case study aims or manages to capture what is geographically specific about them. On the other hand, the thrust of Latour's sociological analysis is to work out a general theory of technology, not to understand a particular phenomena in its context.[142] However, case studies are instrumental for testing and developing general theories. Therefore, to ask how such general theory of technology applies to door systems in particular is a Litmus test of the contribution the case study can make to the social scientific understanding of technology. Such contribution lasts, in turn, on the thesis that hackerspace participants cultivate an alternative engineering culture organised around an alternative understanding of technology.

Latour's study is written in three steps. First, he looks at the problem of walls that are impenetrable for humans. Doors answer the problem because they are walls that can be made open or closed by humans at will. Second, the implementation of doors presents another problem, namely that the door is left open by unruly humans. Grooms solve this problem by closing doors automatically. Third, it turns out that grooms are ultimately unreliable too. The paper note cited above is a desperate attempt to replace the groom by an interpellation to the authority of God that lies in the human heart. Since God's hold over the human heart slackened with the advent of modernity, this ultimately fails. Note that the note on the door is both the starting point and the end of the story, because it makes the initial motivations behind implementing the groom explicit. As Latour and others often point out, when artefacts break down, it is easier to understand how they structure social relations. Finally, as cycles of breakdown and development occur, problems spiral towards complexity so that more and more complicated contraptions are put into place.

Door systems indicating the space state (open/closed) have been developed to address the problem that clubs like hackerspaces have no reliable opening times.[143] Door systems answer the problem by publishing the space state in real time. However, managing the space state presents another problem because members have to be disciplined to toggle the space state when they come in and switch it off when they go out. Open hardware projects to make small scale electronic artefacts – space switches and buttons – positioned strategically next to the entrance solve this problem by offering a convenient way to manage space state. But space state switches are ultimately unreliable too, so that documentation has to be written on how to repair them. The documentation makes it explicit for users how the system works. Eventually, door systems grow to incorporate additional items of the protocol that defines how to properly open

[142] The latter would be perhaps called anthropology today.

[143] Scheduled events and traditional open social nights on Tuesday evenings *are* explicitly set opening times, but in many hackerspaces they are not reliable, and they also represent only a small fraction of the time the hackerspace is actually open to visitors.

and close the hackerspace.

10.5.1 An immanent critique of technology

The parable in 10.1.5 of door systems taking on the tasks of humans is developed further here using the arguments accumulated so far in the light of Latour's work on doors systems. Latour points out that the social relations expressed in the note on the door can be – and eventually do – get implemented in technological artefacts, so that technology comes to play a part in arranging social relations. If the reader looks back at "human protocol notes" on Figure 16 posted at the London Hackspace in its early days, or even the list "Here's how to close up the space when you're leaving" list quoted in 10.1.5 it is evident that they are close relatives to the note mentioned by Latour. Such notes are frequent to appear on the doors of hackerspaces, setting the protocol for the proper procedures of space opening and closing. It is easy to see that almost all items on these check lists have been automated away and maybe even incorporated into door systems in one hackerspace or another. Without repeating the analysis in 10.1.5, let us just accept that for instance all items on Figure 16 (London Hackspace) except toggling space state and putting food away are covered by the Hack42 door system. Ironically, as the space grows there are more and more things to turn off, deactivate, take care of, and so on and so forth. Even in stagnating spaces, members realise more and more problems with the existing infrastructure, problems that call for new items in the protocol. As a result, the rate of automation cannot necessarily keep up with the rate of problems popping up. However, the crucial point is that toggling the space state manually remains an item of the closing protocol for humans in most hackerspaces, even in the face of increasing automation of arbitrary items.

I choose to concentrate on a line of argumentation in Latour's epistemological proposal which is largely consistent with critical Marxist approaches even if it is worded very differently. Let me recapitulate briefly. The author proposes "an extremely simple technique" in three steps (1988b, 298). First, drawing up a list of things humans would have to do if the piece of technology under consideration would be missing from the scenario. Second, another that describes how humans have to behave as users of the same technology to solve the same problem. Third, subtracting one list from the other yields the difference that technology makes in the field of social relations. As the author notes, the result is "a scale balance where tiny efforts balance out mighty weights", that is *technology saves human labour time.* Saving is mobilised in my translation of Latour in all its semiotic ambiguity: positively as absorbing the human labour time of its producers and conserving it for the future, as well as negatively as making a certain amount of human labour unnecessary for its users. The calculation suggested by Latour is the calculation of human labour time saved in technological artefacts. While arguing for putting objects back into sociological analysis, he is advocating a method that takes objects away to reveal the social relations they mask. Marxists would say that the conceptual operation outlined here is called reification – a concept established by Lukács György (1971). Reification is when a set of social relations are turned into an object that stands vis-a-vis its beholder in all its facticity (Latour 1988b; Edgar 2006). Latour advocated the mission of "technologists" to educate the public in general and sociologists in particular to take into account the "missing masses" which are the result of these calculations. Such ambition is not far from the thrust of classic critical theory to provide descriptions of reification in the hope that it will bring consciousness to the masses (Vandenberghe 2008). More precisely, the difference in the two lists measure the characteristic qualities of technologies to reverse (social) forces and fold (human) time:

> We also notice, when drawing the two lists, an interesting difference. In the first relationship (hinges vis-a-vis work of many people), you not only had a reversal of forces (the lever allows gentle manipulations to heavy weights) but also a reversal of time. [...] The first one evokes the past perfect ("once hinges had been installed"); the second the present tense ("when the groom is at his post"). There is a built-in inertia in the first that is largely lacking in the second. A profound temporal shift takes place when nonhumans are appealed to: time is folded (Latour 1988b, 301).

I already argued that time-folds save human labour time by absorbing, storing and repeating it for their users. However, time-folds also spare us the labour of reflection. For Latour, the central difficulty of the mission lies in the "silence" of technology, e.g. that it is much more difficult to interpret for people than literary works where questions of enunciation can be more easily analysed by critics. He mocks sociologists for not being able to follow the rhetorics of technology. In order to translate what technology is doing, it has to be communicated in the form of a script, "strings of sentences ... very much like a programming language" (306). Obviously, hackers generally have less problems understanding technology as a string of sentences like a programming language. Therefore, they are often producing technological artefacts that can be translated for sociologists as a critical reading of technology. Such literature today is found under the rubric of critical engineering practices, for instance.

10.5.2 Non-humans conversing with humans

What is critical in the door system compared to the door-closer then? First of all, hackers do a whole lot to translate their small scale electronic artefacts to human readable language. On the one hand, in the best OSHW/FLOSS tradition, their composition and functionality is often documented on the wiki page of the project. As Söderberg already pointed out, hackers have much in common with critical theorists (2011). On the other hand, even basic implementations of door systems incorporate robots that are logged in to the chat channel of the hackerspace and report everything that the door does. An extreme example is robonaut, the door system bot of London Hackspace, whose interjections often make conversations hard to follow:[144]

```
11:58 <mentar> That’s surprisingly cheap
11:58 <robonaut> Alain Fundi opened the hackspace back door. (Last
seen 18 hours ago)
12:00 <mentar> Don’t trust the website though
12:03 <rophl> plus wouldn’t it take 6 months to print something
with 10 micron layers?
12:06 <robonaut> ag opened the hackspace back door. (Last seen 9
days ago)
12:27 <robonaut> Justin Fishlock opened the hackspace back door.
(Last seen 5 hours ago)
```

These robots turn what the door system does into string of sentences – human readable speech. Turning technologies into actors on the stage of human drama unfolding on the chat channel, they do not let their masters forget what role they play in their histories. The scripts that are inscribed into door systems are explicitly spelled out from the initial notes on the door to the documentation on the wiki and the unceasing reports written on the hackers' screens. In the conclusion of his essay Latour finds the main danger as well as the main promise of technologies in "shifting out" the action from the scene of human actions to the realm of non-humans that falls outside of the usual frame of considerations. For hackers, non-human technological actors remain something to share our lives with and things to relate to explicitly – indeed bots on the chat channel impersonate the door system to bring it back to the realm of language flowing between human and non-human actors: a situation that programmers are intimately familiar with.

10.5.3 Deskilling and work discipline

The door system is a time-fold that does not render human labour completely invisible or utterly unnecessary. The ramifications of this are many and some run deep. As human labour is not hidden from the eyes of users, it is harder to look at a door system as a social fact. The door-closer hides in the corner of the door, trying to become one with its surroundings, while the door system presents itself conspicuously. Neither a theatrical switch or button coupled with colourful LEDs, nor the dangling wires going to a naked PCB look like a finished fact that the user has to helplessly accept without understanding it. Moreover, as hackers continue to insist, the tangible interface of the door system is most often implemented as a button or switch that has to be acted on consciously. It is easy to implement a fully-automatic system of sensors that toggles space state without human interaction – and indeed, Bitlair or HSBXL went down

[144] Chat log. irc://irc.freenode.net/london-hack-space, 2015-05-23.

this road. However, even in those cases there was an attempt to keep humans in control even when giving the job over to a non-human. The members of those latter hackerspaces can use functionality explicitly built into the door system to prevent it from setting the space state. Compare this with Latour's frustration at how many everyday objects fail in the face of the door-closer on his attempts at propping the door open. That is because the door-closer is designed to try keeping the door closed at all costs, which brings us to the question of workers' discipline.

The door-closer – just like Latour's car that keeps beeping if he does not put on the safety belt – has been designed by engineers and employed by people who are in charge of enforcing scripts in the everyday lives of the architectures they plan and run. The head of the university administration or the security manager of a car manufacturer cannot let doors open and safety belts taken off at the whim of their users! The heating bill would spiral out of control and the accidents would cast doubt on the automotive brand if they would trust their users with control of the technology. However, door systems are designed by the same people who will have to live with them ever after, or at least until members get fed up with them. Therefore, they have to be able to bend to the will of their users. As I insisted in Section 9.4 of the previous case study, unfinished artefacts explicitly include functional parts in their technical composition which extend, support and defend their interpretative flexibility.

Here, I add that unfinished artefacts contribute to unfinished architectures whose technological rhetoric is more about emphasising possibilities for change than enforcing behaviours, which lends a sense of agency and ownership to their users. Ultimately, this should not be a surprise given the political economy of hackerspaces, e.g. that they are financed, managed and used primarily by their membership. However, not all social clubs can craft the scripts of technologies to the taste of their users. In order to preserve the democratising potential of technologies developed by participants for their communities the level of technology used has to keep up with the general intellect of the user base.

Latour points to the deskilling thesis – the favourite topic of social historians of technology, as he puts it, presumably referring to works like Noble (1984) – as a good way to theorise such difficulties. He acknowledges that skilled non-humans require non-skilled human users, rehearsing the argument of Noble that the introduction of automation leads to less control of the production by the workers. Indeed, both authors argue that decreasing the autonomy of the working class is often the main reason for the introduction of automation, not the expected increase in efficiency. Noble goes to great lengths to document that what happens in practice after the introduction of automation is almost always a decrease in the control of workers over the production process, while improvements in efficiency do not necessarily materialise or at least very hard to prove quantitatively. Again, door-closers in official buildings require no interaction or understanding by their users. Latour recognises that they do require a bit of maintenance (oiling from time to time) but finds this negligible and notes that it could be performed by maintenance workers who can do a lot more other things as part of their jobs, e.g. they do not have to be full time butlers. Door systems that have buttons or switches require interaction from their users – indeed, the ones that do not are often considered flawed by hackers, since they work independently of the will of the people they are supposed to serve. Moreover, door systems do break down more often than door-closers because they are not designed to be so robust. Even though their designers are held somewhat responsible for their maintenance, any of their users are supposed to be able to fix them when they break. Fortunately, they are designed to be easy to understand and mend. I already recounted the objective reasons for transparency in unfinished artefact, so here I want to stress the subjective components that contribute to unfinished architectures. Along with automation which can easily lead to deskilling (clever machines for stupid users), good hackerspaces and OSHW projects, that is unfinished architectures and unfinished artefacts, teach their users at the same time as they are giving them new tools. Therefore, empowerment happens subjectively as well as objectively, which fend off deskilling through mixing education, research and production.

Conversely, Latour also notes that "An unskilled nonhuman groom thus presupposes a skilled human user." Users have to be educated to use tools and to bend to the scripts inscribed in the machines. In the case of the door-closer people have to take care not to hit their noses into automatically closing doors, and in the case of door systems they have to push the button when they are the first or last to enter or leave the hackerspace. Door-closers employ physical punishment to enforce workers' discipline amongst their user base: they shut the door in the people's faces. Despite all the effort by their human designers, Latour exclaims that "I am ashamed to say that, when I crossed the hallway this fatal February day, the door was open." (1988b, 305) Door systems are designed to employ gentler methods that rely on aesthetic effects of technological rhetoric. In case these fail, they have to fall back to the default way that humans seem to have to replace technology: hand-written appeals to their fellow humans – that is scripts written in their human language of choice.

The technological rhetoric of door systems builds on theatrical elements like the significant forms of buttons and switches, on LEDs installed next to the physical interface that provide feedback to users about the current space state, and the strategic position of the contraptions which are installed near the entrance. Such positioning obviously mobilises the script of light switches that people who lived in a technological civilisation for some time inevitably learn to utilise properly. Of course these measures are not bullet proof. In the same way that people sometimes forget to switch off the lights when they leave home, hackers sometimes forget to switch the space state when they leave the hackerspace. What can be done after they closed the hackerspace? Some hackerspaces like Foulab (Montréal, Canada) and the LAG hacklab (Amsterdam, The Netherlands) have big signs that shine bright outside of the hackerspace, reminding hackers on their way out that the space state is still OPEN. This is an opportunity for absent minded hackers to turn back and set the correct value. Since many hackerspaces have a very active chat channel with continuous conversation and updates announcing the space state. It happens sometimes that online participants find out in one way or another that the space is officially open while there is in fact nobody there. Other times hackers get home and realise they made a mistake and they left the space state on. Therefore many door systems provide an online interface that overrides the settings of the tangible interface, and many of these systems can be controlled through commands to the door system bot on the chat channel (others work through a web interface or even more exotic ways). Again, the principle is not to make something unbreakable but to provide ways to recover from failures. Such a design principle is derived from the strong belief of hackers that every system can be broken, without which there would not be independent security research, penetration testing industry, or reverse engineering. There is also peer pressure applied to users – members in good standing are expected to remember toggling the space state appropriately and they are told off when they forget it. To summarise, the three elements that discipline users to do the right thing with door systems are *technological rhetoric*, *fail-safe mechanisms* and *social norms* which are actively enforced by the members of the community. The three come together when closing the hackerspace becomes part of a ritual that is performed as a routine, merging artefacts that are necessary accessories with human actions that are socially meaningful.

10.5.4 Social groups addressed by door systems

The time compression that a door system performs can be thus translated into the strings of sentences studied by Latour. The lengths and sophistication of that list depends on the exact implementation of the door system. For the sake of argument let us take a typical implementation that includes common elements from the artefacts seen in 10.1. Toggling the space state would update the website with an image announcing that the hackerspace is open, publish a file on the Internet using the SpaceAPI format, update a Twitter account and make the bot on the chat channel change the topic to something like “Welcome to Technologia Incognita, we are open.” Then a member who installed the SpaceAPI Android Application to their smart phone would get a pop-up message about the event. Another member would get a notification from their operating system because their IRC client (the software they use to participate in the chat channel) would be configured to alert them on space state changes. The one action of pushing a button would thus unleash a chain of events. The notifications in the first sentence would be *pushed* actively through different media, while the two latter examples would work through a *pull* model where the application actively polls for changes. The second type of notifications would not have to be implemented in the door system itself, because their operation is completely independent of the door system mechanism – they are merely reading the SpaceAPI file or the chat channel periodically. Using the standard interfaces, users could introduce even more time compression according to the their preferences without interfering with the system. The one button therefore sends reverberations across the World Wide Web, Twitter, IRC, reaching an Android phone and perhaps an OS operating system running on an Apple laptop.

Just like somebody breaking a wall with a sledgehammer and rebuilding it with mortar and bricks instead of using a door, the member who enters the space could in theory do all these things herself: edit the source code of the website to change an image, write a file on the web server for the SpaceAPI, send a message on Twitter and make an announcement on IRC. As explained before, she would not even have to worry about the interfaces that *pull* space state announcements, since they are operated by the users who consume the messages. However, only a small number of hackerspace members have account over all these systems. This is aggravated by the fact that similarly to vegans in the squatting scene who abstain from eating meat for political reasons, many hackers boycott technologies they deem oppressive. On the one hand, asking all hackerspace members to register on Twitter would meet with serious opposition in any hackerspace I am familiar with, because the social media monopoly is considered a surveillance machine whose control is out of reach for its users and which is structured in a way that makes any deliberative conversations completely impossible and which encourages people to repeat what others said endlessly: the very definition of a political echo chamber. On the other hand, using IRC conveniently requires basic system administration skills and access to a shell server, a machine which is connected to the Internet non-stop. All hackerspaces have active IRC channels which proves that such level of access to material infrastructures and such a level of expertise is very common amongst their members, but it is totally sure that not all members fulfil both requirements. This is evident from online conversations where some users relay for members who are "not on IRC". Such compression has multiple consequences to the social dynamics of the hackerspace.

First, the various components that do the various parts of the chain reaction require various kinds of expertise which are likely to be mastered by various members of the hackerspace. Developing a sophisticated door system is a never ending labour where collaboration between members is not merely desirable but often absolutely necessary. In a way the door system accomplished the exact opposite of the door-closer in Latour's story: instead of relieving humans from labouring through the introduction of non-human actors, it actually gives them an excuse for never-ending technological toil. However, as long as the production of unfinished artefacts like the door systems is a certain significant byproduct of the specific sociality that unfinished architectures like hackerspace cultivate, this is not detrimental but beneficial for social life. As I argued before while presenting the integration projects growing up between different door system implementations in 10.3, door systems extend, exchange traits and eventually combine into large technological infrastructures. In turn, these infrastructures become constitutive to the everyday functioning, gate-keeping and identity of the hackerspaces scene as a whole. Latour notices a similar tendency regarding the increasingly more complicated solutions to the simple problem of doors when he writes that:

> You seem to always need more and more of these figurated delegates aligned in rows. It is the same with delegates as with drugs; you start with soft ones qand end by shooting up. There is an inflation for delegated characters too (Latour 1988b, 305).

The technological imperative of Jacques Ellul (1964; 1980) in general or the digital imperative of Wyatt (2008) in particular are at work here. Modernity ties progress tightly together with technological development, so that technology seems to move to inevitably to new levels of abstraction and complexity. Late modernity's obsession with digital technologies presents them as a condition of survival at least for those who have to sell their human work time on the market. However, there are many points of resistance to such a trend too, not least within the wide variety of peer production phenomena.

Once again, the twist that door systems bring to these imperatives is that the peer production practices proliferating around them go beyond the instrumental rationality that characterises humans' relationship to non-humans (technology and nature). Söderberg termed these resistances based on everyday practices and deviant socialities *play struggles* (Söderberg 2011). I believe that these aspects of peer production are more historically significant than optimising economical relations through more efficient exchange, as proponents of transaction cost theory would have it. One hacker exclaimed with pride that the mission of hackerspaces is "to find complicated solutions to non-existent problems" – an ironic comment on a native definition of a hack as a simple solution to a complicated problem.

Second, Latour warns his readers in the manner of Winner (1999) that the door-closer discriminated against furniture movers because it makes it cumbersome to move large objects through the door, as well as grandmothers because it requires a stronger force to open a door equipped with a door-closer than a normal specimen. Similarly, even if the door system inspires and enables collaboration while coordinating and collecting a diversity of technologically inspired participants – as argued in the next point – as part of the same gesture it also discriminates against some users. The limits of participation circumscribed by door systems culture is not difficult to see, so that conclusions are rather straightforward. Using the door system as the primary coordination mechanism assumes a highly educated, mobile, flexible workforce in the inner city with practically constant Internet connection through relative smart devices. Single mothers are arguably put into a disadvantage by such a system because most of them have to plan ahead to a couple of days, so that they cannot take advantage of the spontaneity that is afforded to them by a door system. So are residents of the surrounding towns, since they cannot just "pop in" to the hackerspace. The door system affords these freedoms to people who already have certain privileges: it fits into their form of life.

Hackerspaces do rely on secondary coordinating mechanisms such as organising open social evenings typically on Tuesdays. Exposing the limits of coordinating through door systems, the latter events do bring together a wider variety of people, many of which are not able or not motivated to visit the hackerspace on other occasions. However, at the end of the day being an active member of the hackerspace means much more than coming together on Tuesdays: it means taking advantage of the 24/7 access to the hackerspace and to work on projects with other members at odd hours.

Finally, door systems allow the participation of a relatively wide range of people with different technical preferences and different levels of expertise. We have to remember that the fact that hackers are invested in and knowledgeable about technology does not mean that they love all kinds of technology: some they love dearly, and some they hate with a passion. Therefore, it is important to find solutions that allow for diversity and at the same time thrive to hammer out common platforms like the SpaceAPI which constitute an acceptable middle point for people and technologies to meet. I develop these arguments further in 1.6 where I look at how door systems help hackers to match their various and variable schedules to each others'. Here it is enough to note that a door system allows reaching a multi-media audience with a single movement of the hand.

10.5.5 The encoding problem

What can be said about unfinished artefacts using the "simple technique" and "coherent vocabulary" that Latour (1988b) is proposing for understanding the society/technology nexus based on the work of Akrich (1992) that they also published together as A Summary of a Convenient Vocabulary for the Semiotics of Human and Nonhuman Assemblies (Akrich and Latour 1992) then? Compared to mainstream technologies like the door-closer that Latour makes the centre of his study, unfinished artefacts are easy to *describe* because the process of *transcription* is made explicit to users, but beyond that they also make it as easy to *subscribe* to their scripts as to *des-inscribe* them. The first part of the sentence applies to OSHW by definition since what makes something "open source" is exactly the publication of "a string of sentences", e.g. the script of the artefact. That helps a lot in changing the artefact but does not exhaust the wide range of political possibilities for the democratisation of technology that are involved in hacker practices. The second part of the sentence after "but" emphasises how unfinished artefacts go beyond the formal OSHW requirements to encompass properties of the technical composition. *Subscription* should be easy because systems are simple, while *des-inscription* – which is essentially Latour's rediscovery of the *hacking* paradigm[145] – is easy because the technical composition is built on open standards, uses off-the-shelf components and does as little blackboxing as possible (naked PCBs and dangling wires is the trope that I was using to describe these design decisions that result in inviting technical rhetorics).

[145] Popular definitions of hacking include using something in a way that it was not intended to be used, making artefacts do something else then what they were designed for.

A more detailed *description* of how shared values of an engineering subculture are translated to materiality is found in Section 9.4 in the first case study where I use the SCOT language to argue that unfinished artefacts include specific functional components whose purpose is to fend off stabilisation and closure while defending the interpretative flexibility of the artefact even after it has passed from designers to users. However, it is crucial to remember that neither r0kets nor door systems are worth much outside their specific milieu of the hackerspaces. Unfinished artefacts are tied to unfinished architectures which support peer production practices through making expertise a limit of participation. In Latour's terminology (this time adopted from the biologist Waddington, 308), expertise of the user base is an important element of the *chreod* of the artefact that comprises all the other things that have to work for this one artefact to be useful. In other words, the freedom afforded by unfinished artefacts is best enjoyed with a baseline of technical expertise. As a popular saying sometimes attributed to Alexander L. Haiut goes, "Contrary to popular belief, UNIX is user-friendly. It just happens to be very selective about who it decides to make friends with."

All in all, most of these properties of the script prescribed by unfinished artefacts like door systems can be explained by the fact that they were developed by hackers for hackers. When Latour searches for structuring factors in society and finds that machines are implicated, he muses that "[e]ven if it is now obvious the missing masses of our society are to be found among the nonhuman mechanisms, it is not clear how they get there and why they are missing from most accounts." Of course, he finds that engineers have put them there, and the reason that they are missing from sociological accounts is that unlike authors of texts like novelists, authors of technological artefacts like engineers hide the *translation* of the action, *shifting* their characters out of the scene. Only through *describing* and thus bringing them back to the centre of the discussion can we address the challenges of what Latour calls modern societies in general (310) or late capitalist societies in particular (302). This is all the more important because as he argues – in a veritable rhetorical move of critical theory – the power of machines over our lives is not only invisible but also pervasive:

> If, in our societies, there are thousands of such lieutenants to which we have delegated competences, it means that what defines our social relations is, for the most part, prescribed back to us by nonhumans (310).

Here they seem to agree with Feenberg, except perhaps on the point of where the power lies. Feenberg's position is ambiguous between blaming the management or the engineers, as both could be considered the "masters of technical systems", but he considers the power of labourers in making technical decisions the key point for making technology work for them. Democracy is thus achieved by calculating the architectural effects of artefacts into discussions and decisions:

> Technology is one of the major sources of public power in modern societies. So far as decisions affecting our daily lives are concerned, political democracy is largely overshadowed by the enormous power wielded by the masters of technical systems. [...] Marx [...] claimed that we will remain disenfranchised and alienated so long as we have no say in industrial decision-making (1992, 301).

Putting two and two together, Latour's own position is not as far from this as it may seem at first sight. Even though he puts artefacts in the centre of the discussion and proclaims that considering the missing masses accounts for the democratic deficit in capitalism, he does feel it essential to remember that the inscription is done by engineers and ordered by managers, whose preferences lend the artefact a certain agenda. Thus Latour and Feenberg meet on the shop floor of workers' control over the means of production, and that is exactly the point where peer production practices of small scale electronic artefacts in hackerspaces becomes interesting. Hackers are mobilising engineering expertise to translate, displace, transcribe, inscribe or encode (in Latour's many simple words) human action to durable material artefacts – but they are doing it on their own according, for their own purposes and using their own shared machine shops. What's even more exciting from the point of view of democratic devices is that these labs are open to the general public and thus encourage participation in the study, development and application of engineering expertise. Finally, as we have seen in the preceding chapters, hackerspaces are not restricted to the practical expression of technological creativity but cultivate a specific sociality that includes a lot of socialisation, much of which is debate over technology and the specific social consequences of specific technical decisions – perhaps the very debate STS scholars from Feenberg to Latour would like to see. If the "masters of technical systems" are the system administrators themselves then they will obviously make different systems which are better – at least for them! – than if managers determine the requirements and provide the social and material conditions for production. The question whether system administrators who become hackers could serve the interest of dominated social groups in general is a difficult one – but to be fair with them, at least they feel as dominated by technology as anybody in this society.

The promise of certain social groups learning about their own history while designing and manufacturing their own artefacts is that they will not become solely oriented towards instrumental rationality that is used as a force of social domination through structuring the actions of its users invisibly and incontestably by manipulating the forces of nature. It is not a novel idea, but hard to realise without three key components that a social group has to posses. First, developing an alternative engineering culture that allows for conceptualising a relationship to technology that is different from consumer goods. Second, developing technological expertise in the area of teaching, researching and manufacturing unfinished artefacts. Third, developing the fixed capital, e.g. the material infrastructures or unfinished architectures for the realisation of such activities. It is easy to see that such a social group has to be relatively privileged in order to do all that, but also comparatively pissed off about recent developments in the area of science and technology to stride off the trodden path and trace out alternative trajectories of technology. I have argued that historically the relative autonomy of the hacker scene served as a class formation mechanism through which participants realised these conditions. Here I showed that door systems like unfinished artefacts share many of the properties of door-closers as consumer goods but there are crucial differences in terms of the *translation* of the *script* some of which are politically significant.

10.6 Organising flexible working spaces through techno-social networks

After the low-level analysis of the door system in the previous section, the next step is to investigate what sense these actions make in the context of the social transformations that characterise late capitalism. The high-level analysis can account for some of the automatisms and default behaviours we see on the ground, while highlight how everyday practices that may not look subversive at first sight go against the grain. Methodologically, then, the sociological analysis provides the background against which the ethnographic results stand out in their particularity. So far the accent of the analysis have been on the internal techno-social dynamics of the hacker scene, but here hackers are considered mainly as general late capitalist subjects who struggle with their specific late capitalist problems.

Therefore, the question is **how participation in hacker clubs is organised in response to changing social conditions, through conventions and technologies?** Following the question, the three units of analysis to address are the changing conditions in society as a whole, the hacker clubs as organisations in the middle range, and finally the case study of techno-social means through which participation is organised. As introduced in the theoretical framework (Chapter 3), the theories that stand in for the changing conditions in society are those of reflexive modernity (Beck, Bons, and Lau 2003; Beck, Giddens, and Lash 1994), neoliberal hegemony (Harvey 2005), the network society (Castells 1996; Castells 1997; Castells 1998) and cognitive capitalism (Boutang 2011). In the next section I recap the basic tenets of each and highlight the elements that are the most relevant in respect to hackers, hackerspaces and door systems.

10.6.1 Transformations in the world of work

The network society refers to the transformation of capitalism from the 1970s on which constitutes a fundamental change in social conditions. Right at the dawn of the era Touraine (1969) and Bell (1973) theorised this phase of capitalism as *postindustrial*, a term which have been qualified many times by scholars in the last decades. More lately, Boutang (2011) addresses similar transformations as *cognitive capitalism.* For the moment, we focus on Castells' concept of *network society* (2014). All accounts agree that at the very least in the wealthier economies of the West, service sector and especially technology and knowledge intensive productive activities moved to the core of the economy, both in qualitative and quantitative terms. Automation based on micro-processors is one of the most pervasive change affecting all sectors. For the study of hackerspaces and technocratic governance it is crucial to realise that there is an engineering problem at the heart of these transformations.

Castells identifies the architecture of computer networks (not the computers themselves) as the decisive enabling and structuring factor in the transition to the network society. In the field of economy, competitiveness is based on the capacity to generate knowledge and process information effectively. The rise of the network enterprise follows, which results in the transformation of employment and labour: the world of work. The latter is the locus where large-scale changes enter into the life-world of hackers, and therefore the focus of this section.

Castells emphasises how actors "strategically reconfigure the geography of networks according to their interests" (2014 Chapter 2, 3). While multinational corporations employ only a small portion of the labour force, they are central to the economy. 260 million workers of the 3 billion global labour force work at 78.000 multinational corporations, yet produce 40% of global GDP and two thirds of international trade. At the same time, the informal economy is a more obscure but nonetheless important element. Computer networks enable many aspects – such as the finance sector which plays a structuring role – of the global economy to function in "real time" (Ibid. 16).

Ensmenger (2010) argues persuasively that even if technology professionals are not in leading positions, programmers and system administrators are essential since they put processors and networks to work. Their hold over technological expertise as well as the actual work of implementing managerial decisions means that they can in many cases influence decision makers through their advice, or alternatively, bend and even bypass managerial decisions while they are "executing" them. These observations show that highly skilled technology workers are not only well-paid but have a relative autonomy on the labour market. Today, secret services have to go to considerable lengths to find enough hackers to join their ranks, so much so that the FBI is advertising as a marijuana friendly work place (Levinson 2014; Cherney 2014). As the Snowden leaks proved, employers can never be sure of the loyalty of their workers. These problems for employers come from the fact that in many important areas the "industrial reserve army" (Marx 2007 Volume I., Part II, 699) is not actually very large. Recent discussions and debates over the quality and quantity of STEM education in the United States show similar concerns. Altman (2012b) notes how even the US military is trying to enroll hackerspaces into their recruitment and innovation workflows in order to attract candidates and develop technologies. These disparate facts testify to the relative autonomy of the hacker scene and hint at how hackers can have a comparatively strong position in negotiation with their employers without formal union organising or traditional wild cat strike – even though sometimes through an innovative use of slow downs, work-to-rule practices and sabotage.

Access to know-how is essential since it is the main productive force for economic development. Productivity growth depends on technological and organisational innovation. According to Caldas, David, and Ormanidhi (2005) these two factors have to go hand-in-hand in order to be effective. These notions reverberate strongly with the analysis of hackerspaces as unfinished architectures (organisations) where unfinished artefacts (material objects) are researched and produced. From the point of view of capital, hackerspaces are uniquely positioned since they put organisational innovation in the service of technological innovation. At the same time, they are organised by hackers, for hackers, and their sole purpose is to facilitate hacking. The two cases here can be read as studies of social innovation which combine organisational innovation with technological innovation.

Analysing trends in the management literature of the last decades, Boltanski and Chiapello find that the project becomes the real operational unit, which needs specific kinds of humans and non-humans to operate (Boltanski and Chiapello 2005, 104–105). These results broadly align with Castells' results. Independent consultants who are flexible and self-programmable are key to their successful implementation. *Self-programmable labour* refers to workers who are not only highly educated but also fast learners and good in adapting to changing requirements. Their verso are *generic labourers* who can be easily replaced by similar workers elsewhere, or alternatively by machines. In this sense we can talk about self-programmable and programmed labour, and note that the more sophisticated part of such programming is often undertaken by self-programmable workers. It is evident that hackers fall in this latter category: they are programmers in the double sense of the world. As Dutton (1999) explores in detail, these dynamic business models which culminate in the *virtual organisation* depend on the Internet, intranets, and other forms of computer networking. For these reasons, project consultants and technology workers are in high demand: in Western Europe and the USA of the mid-2000s, 20% of the market demand was not met.

Structural changes have detrimental effects which hit especially the relatively disadvantaged portions of ICT workers. Faster technological cycles and ageist culture mean that older workers who find themselves on the job market have a hard time finding employment again, and in the absence of an effective training schemes this can result in "pockets of long-term unemployment" (Quintana and Mora 2012). Unionisation plummets and consequentially the weaker workers lack any form of protection: collective bargaining power declines. Furthermore, even under such dramatic changes, the changes in the relations of production lag behind the more rapid changes in the forces of production, leading to specific problems which arise between economic organisations and workers. Namely, workplace surveillance, new occupational pathologies from computerised work and institutional rigidity hamper both productivity and job creation. Discontent with management, technology leadership and working conditions is recurring topic of discussion in hackerspaces.

Again, hackerspaces bring a model for the self-management of self-programmable programmers through organisational innovation which facilitates faster rates of technological innovation. In this model the network enterprise is stripped down to its bare bones. These essential elements are idealised in the concept of *lean* in contemporary management literature (Furr, Dyer, and Christensen 2014; Ries 2011; Humble, Molesky, and O'Reilly 2015). The difference between lean business organisations and hackerspaces is that the former creates value directly for customers as well as indirectly for shareholders, while the latter is the pure self-valorisation of the workers. The notorious hacker phrase "for fun and profit" attests to that. As we will see, governing hackerspaces members value two things first and foremost: the social well-being of members and technological experimentation – everything in the hackerspace is a mere corollary to those two points. For instance while pure self-valorisation often leads to spin-off companies emerging from the hackerspace, spin-offs are treated as just another aspect of life rather than a central fetish like in the lean startup culture. Another genre of shared machine shops are the accelerators[146] presented in 8.4 whose core mission is to create lean startups.

Interestingly, despite the shortening cycles of innovation, the overall impact of automation is still less hours of work per worker, which can have either of two effects. One options is that while a significant portion of the labour force is out of work, the employed have to put in many more hours of work than before. The other option is that workers in general have to work fewer hours in order to attain more or less the same wage. Of course the net effect is a mixture of these two, but there seems to be a work time somewhere liberated by the introduction of new technologies. Peer production models where *free time* is mobilised for value creation and often captured for capital accumulation are capital's answer to this problem. Consequently, hackerspaces can be interpreted both as the reappropriation of free time by workers for socially useful production (Smith 2014) and on the contrary the recuperation of free time by capitalism for eventually turning such *free* work into labour (Terranova 2000). In the final analysis, the meaning of the trope "fun and profit" is conflated: fun and profit become one.

10.6.2 Late capitalist subjectivities

Giddens argues that in late modernity there is a pressure on people to construct their identity as individuals because society does not present them with a ready-made role that they can grow into (1991). Identity is constructed through a narrative about oneself, a process full of existential uncertainty yet ripe with

[146] Alternative terms exist for the same type of institution, such as incubators or hubs. Even some co-working spaces provide similar services. Local and national governments often support these with the hopes of boosting the local or national economy.

the potential of self-fulfilment. Similarly to other cultures and subcultures, the hacker scene offers a comprehensive package for participants about who they are and offers an environment for performing their hacker identity. Through the ritualistic elements of door systems I was emphasising how the construction of small scale electronic artefacts can be an entry into the hackerspaces scene as a way for the individual to perform and therefore produce her hackerdom. The term used in the common platform of leading European sociologists – reflexive modernity – emphasises that late modernity does not stand against traditional pre-modern any more but challenges the institutions of modernity itself. One consequence is the increasing uncertainty that both institutions and individuals have to face while they are often on a collision course. In line with the thesis of reflexive modernisation I have argued in Section 7.3 that the hackerspace as a social formation challenges modern institutions which compartmentalise human activities such as education, research and manufacturing.

In the forthcoming Section (10.7) I argue that the space state as a concept put into work by the material infrastructure of the door system challenges the modern institution of the opening time and its material manifestation in the time clock. Just as the space state and the door system constitute the subjective and objective aspects of a specific post-industrial regime of work discipline, the institution of opening time and the time clock are the subject and objective aspects of the industrial regime of work discipline. Once again, in accordance with the thesis of reflexive modernisation the door system represents a challenge to modern institutions which is the result of surfacing contradictions between the individual and the institution as much as it is a critique of the separation of individuals. I argue that the door system does not merely reflect the contradictions of reflexive modernisation – even though it does that pretty accurately too. Additionally, it also articulates a synthesis which is in itself an interesting social innovation that can be conceptualised as a comment or contribution to the theory of modernisation as a historical process.

Zygmunt Bauman is another theorists of second modernity who argues that reflexive modernisation results in novel tensions around the figure of the stranger as somebody who is present but unknown (Bauman 1991). Even though market exchange and consumer society is based on differences that are sometimes actively proliferated by capital itself, difference also marks the limit of order making efforts that characterise modernity since the beginning. In Bauman's terminology, the history of solid modernity was the history of order making efforts that traded individual liberties for security, while liquid modernity does the opposite. The reason for the inversion is the ideological crisis of the welfare state which failed to provide a meaningful life to citizens. Reflexive modernity is a historical process where people reflect on the problems of that first modernity. However, the way that the expansion of personal liberties have been implemented resulted in extreme individualisation that now presents new problems. Liquid fears of the unknown stranger loom over the population. While Beck's risk society concept emphasises how these uncertainties about the future and fears of the unknown are leverages and negotiated (1992), in Bauman's reading these become pervasive existential concerns. Terrorists, paedophiles, serial killers are dispersed within the population and almost impossible to eradicate or even identify, while other predatory ills such as domestic violence, AIDS, government surveillance mean that anybody can become a victim – or even perpetrator of crimes. After the exterminations of Jews which was the conclusion of solid modernity, now it is immigrants who serve as the focus of such fears. Bauman argues that instrumental rationality, division of labour and chains of command are responsible for the systematic abuses caused by liquid fears. It is easy to see that the general economy of Bataille (1991), the self-selection of tasks, and the horizontal networks of communication that characterise peer production practices address these same deep routed uncertainties in a different way.

The door system is a concrete invention, organisation innovation and material infrastructure that reconceptualises the relationship between an established community and the stranger. Organisations with opening times understand themselves and they are understood by others as providing a service to the general public. There are a very well defined set of expectations that go with such a mission. Whether such places are operated by the organs of the state, capital or civil society, they are supposed to implement some basic tenets of the market: bureaucratic rationality in general and nominal equality of the participants in particular. It is expected that they function in a reliable manner similar to classic Weberian bureaucracies of rational-legal authority. These extremely alienating requirements mean that any worker who sites behind the counter should provide the same service to any consumer who stands in front. Rationalisation works with clearly defined rules and predictable operation which are dehumanising, yet promises a universal equality that preserves basic human rights in return. It is not very difficult to understand that such scenario does not leave much space for personal expression and self-fulfilment within the organisation, neither for clients not for consumers. I have argued that these criteria coincide with some of the basic tenets of the market because the market is the place which seeks to create enough trust for buyers and sellers to meet and exchange goods without having to know each other personally and establishing a quantitative relationship. Of course peer production theorists want to extend the same logic to all spheres of life when they argue in terms of transaction costs theory.

Clubs like Gentlemen's Clubs are the flip side of the coin. They are explicitly *private* clubs, exercising the freedom of association granted citizens in modern societies. They are part of civil society but, crucially, they are *not* providing a public service. In fact as the discussion in Section 7.6 shows, they have been invented to counter the dehumanising and alienating tendencies of modern societies, with a special emphasis on the impersonality of urban life and the individualising tendencies of market participation. Clubs are obviously members only and new members have to be vouched for by existing members and the membership have to vote on the inclusion of them. Moreover, criteria are not impersonal at all: in fact classic clubs have two kinds of requirements. One is the formal or external definition of the club as for instance the club of Scottish businessmen in London. The other concerns the social interactions inside the club so it can be called informal or internal requirements. The latter regulate the character and the manner of participants, for example the Eccentric Club's motto *Nil nisi bonum* which is accompanies by a manifesto. New members are vetted against both requirements. All these rules are obviously in diametrical opposition to rational-legal forms of authority.

Door systems obviously play a *key role* in the gatekeeping practices of hackerspaces. The social norm of the space state is inscribed into the organisation through the space system. As a material residue of critique that expresses the social contradictions of reflexive modernity, the door systems present a model that diverges from both the impersonal market model of a civil society association which provides a service to the general public and the classic clubs which shelter their members from the responsibilities of the market by catering exclusively for their personal needs. Even in the case of a traditional volunteer-run NGO, activists are sitting on site during fixed opening hours because they are there for others to come. Inversely, classic clubs are founded to shelter their members from the responsibilities of caring for strangers. The dynamic opening time of hackerspaces which is practically possible only because the innovative door system implementations allows for a synthesis where members can attend the laboratory when they please yet they are compelled to welcome strangers when they do. Therefore, members go to the hackerspace because it is a meaningful activity for them personally – not least because of the opportunity to meet uninvited others who are broadly aligned with their interests. In this way the organisation is structured through material artefacts and social conventions in a way that is both subversive of institutionalised modern interpersonal relations and at the same time rewarding in itself for both members and visitors, so they can meet as peers.

The collaborative culture, absence of the chain of command, the self-allocation of tasks and the networked organisation of peer production projects is evidently conductive of such togetherness. Of course it is not all roses either: meeting as persons can be easily uncomfortable for both sides, and guests who prove not to qualify are surely shunned or sometimes sent away. Members who mistreat guests are also routinely repressed and even excluded from the hackerspace. Meeting as persons on the ground of the hackerspace as a kind of third space is an opportunity to forge personal ties and test resonances. While most members of the general public have very good experiences on their visits to the hackerspace, inexperienced guests often fail to recognise the cultural context they enter and expect to consume services that are provided to the general public. In practice this means that they expect a short way based on the market logic to take advantage of the technological expertise of members. However, as the production and process of the construction of door systems shows clearly, hackerspaces work through open participation in the exercise of expertise. In other words, production is based on a peer production model rather than on a service-product model.

Visitors are in fact allowed even in the classic clubs as long as they arrive as the guest of a member, since the member herself is supposed to be responsible for the behaviour of the visitor. This is not that different from the way hackerspaces treat visitors. Many hackerspaces have social rules that explicitly say that guests can only be in the space as long as a member is supervising their activities and of course such rules are inscribed in material infrastructures because only members have keys to the hackerspace. What is the difference then compared to the guests in classic clubs and how door systems articulate such difference? The difference is that social rules of hackerspaces also state that members in good standing are expected to take care of visitors who drop in to the space and welcome them as guests. So the subtle but nonetheless crucial difference from the operation of classic clubs is that there the guests can only *arrive* accompanied by members, which implies that the member and the visitor knows and trusts each other before they enter the space. In effect a hackerspace is completely open for *strangers* whereas a classic club is not at all. In such capacity the techno-social rules of hackerspaces as unfinished architectures directly address the concerns Bauman raised about *strangers* in the late capitalist era of reflexive modernisation.

Even though hackerspaces interpreted in the context of reflexive modernisation look like an anomaly in contemporary capitalism theories of the project order from Boltanski and Chiapello (2005), the network enterprise from Castells (1996), or the ethical economy of peer production theorists Arvidsson and Peitersen (2013) account for the way in which such hospitality is a beneficial or even required condition for survival in the political-economic life of late capitalism. There are three essential points to take note of here. On the one hand, hackerspaces are the product of society and they are determined by the material, social and technical conditions on the ground. They are not a novel or disruptive phenomena but integrate into structural changes in late capitalism in a systematic and organic way. While such a notion seems like a commonplace it is necessary to spell it out explicitly since it contradicts most scholarly writing that has been published about peer production in general and hackers in particular in the previous decades. On the other hand, hackerspaces are not only part of history – they are at the forefront of history. The very reason why they appear to be an untimely intervention into the ongoing social process is that they are determined by developing tendencies at the cutting edge of contemporary capitalism. I have already argued that hackerspaces can occupy such a position thanks to the social position of hackers as one of the most privileged part of the working class, combined with their historically contradictory efforts at institutionalisation while building a relative autonomy vis-a-vis the state and capital.

To those two now we can add a third element: vis-a-vis the practices of the liberal-bourgeois public sphere of Habermas that are intimately coupled with rational-legal authority of the Weberian kind. Ultimately, hackerspaces gather the most strident tendencies of contemporary capitalism in the hands of a social group with both objective agency (resources) and autonomy (culture). Thus they rearticulate the contradictions of late capitalism in a configuration that can be seen as particularly progressive, or at least bear political potential especially in the areas of workers' self-management, democratisation of expertise and subverting instrumental rationality. Door systems are the material residue of that critique in the sense they explicitly mark out the space of strangers in the functional composition of the hackerspace. They allow for workers and non-workers to meet outside of the confines imposed by the market-based criteria of instrumental rationality in order to share technological expertise with each other. Crucially, door systems are not rational instruments constructed according to mainstream engineering standards themselves that would simply allow such an alternative form of sociality to function correctly and efficiently. Rather, door systems themselves are a byproduct of that same sociality and therefore constructed according to its principles. In that sense they also embody the seldom achieved ideal of reflexive modernity: the harmony of means and ends.

10.6.3 Geographical shifts

Another aspect of the transformation can be analysed through the lenses of human geography. Despite early academic prophecies, working from home (telework) did not become a reality – instead, the multi-location of work places prevailed in the changing geography of the firm (Goddard and Gillespie 1986). In other words, not only the information processes but also the workers themselves became mobile and distributed. As we will see later on, the hackerspaces are a privileged geographical location suited for such multi-located workers to pursue their productive activities.

Putting the two last points in the context of post-industrialism prepared the understanding of the hackerspaces as an urban phenomena. Postindustrial capitalism leaves in its wake vast unused industrial properties, often at the close edge of the city. At the same time, work gets dispersed across multiple locations and between multiple networks of people. However, people still need to be rooted somewhere (Putman 2000) and the hackerspaces provide a matrix of possibilities to accommodate these dispersed networking needs: spaces where flow-driven flexible workers can band together. The argument that hackerspaces can constitute a place of belonging amidst the estrangement and alienation that is the staple of contemporary urban life is made in 7.4. Furthermore, Chapter 8 takes other genres of shared machine shops one by one. These other genres of shared machine shops more or less fit the conditions identified in this section. However, for different reasons they cannot function adequately as third spaces as defined in Oldenburg (1989). While these reasons are argued in detail in the mentioned Chapter, alcohol consumption is a quick and easy way to make the case. Drinks characterise third spaces where participant can develop a belonging and hackerspaces are unique amongst the great variety of shared machine shops in that they usually include fridges with beer – even if hackers often prefer Club Mate.

These geographical changes are complemented by changes in work patterns. In general hackerspace members can be divided into three groups in terms of work patterns: the highly educated unemployed who have in some sense too much time, middle class information workers with a bit of time and money on their hands, and elite high-tech workers whose case is the most complicated because they can easily negotiate their employment conditions with their employers. Of the latter group, some spend only a few days earning a considerable wage, and dedicate the rest to the hackerspace, while on the other end of the spectrum others dedicate themselves to their professional carriers entirely, so that they have no time to actually go to the hackerspace. Even so, all of them play distinctive roles in the organisation and upkeep of hacker clubs, as detailed in Section 7.5. Whatever kinds of people attend the hackerspace, most are increasingly flexible in their own ways, and they have to negotiate these flexibilities and constraints in their club. That is why hackerspaces are theorised here as flexible working places.

10.6.4 Diversity and discrimination

I have argued in Section 7.5 that hackerspaces differ from other shared machine shops like co-working spaces and profit-oriented Fab Labs in that their open organisational architecture allows for much more diversity of social backgrounds. The various groups of people attracted to hackerspaces play different roles in the ecosystem of the organisation, yet they are all effected by the precarisation and flexibilisation of digital labour and respond to them collectively through their practices of self-organisation. The common thread in their responses is that hackerspaces enable them to work more effectively, learn more quickly and have much more fun than pursuing their various activities individually. That is not to argue that the hackerspace is an ideal world of reconstituted communities. As Sophie Toupin shows (2013; 2014) from an intersectional feminist perspective, social exclusion is a very real factor in hackerspaces, manifesting in manifold social conflicts and engendering resistance, including the founding of alternative initiatives driven by women and transgender participants who decide to take a separatist stand. In Section 10.5.4 I highlighted particular ways in which the flexible organisation of togetherness through door systems discriminated against certain social groups - single mothers who have to plan their time in advance and with much care were one example.

10.6.5 The self-organisation of digital labour

I relate my observations to Franco Berardi a.k.a. Bifo, Jacquemet, and Vitali (2009) about their experience of organising precarious creative workers in Northern Italy in the beginning of the second millennium. Bifo and his friends were organising workers before in the golden age of Autonomia in the 1970s. Today, under the shifting existential conditions of what he calls cyberspace and cybertime – the fluidity of spatial and temporal rootedness – the organisation of labour faces different problems than at the time of the classic workers' movement. The specific challenge they face in this new context is establishing the basic conditions for practical solidarity: shared time and space. In contrast to factory workers who have the shop-floor as the baseline of class formation that constitutes a shared space and time, the flexible, self-programmable labour force of contemporary capitalism is so dispersed that it is a challenge even to find a common place and time where they can meet in their bodies and be together. This is the question raised by the political-economical condition of the flexibilisation of labour which the hackerspaces answer. And they answer it through a specific configuration of human and non-human actors, the unfinished architecture of a network organisation which is less a parallel, more an antidote to the network enterprise. In essence the hackerspace (as unfinished architecture) provides the shared space and the door system (as unfinished artefact) provides the shared time for coming together.

There are many different social groups which are affected by the precarisation and flexibilisation of labour. However, not all of them can maintain a sustainable social spaces for their own use without hegemonic institutional support or without accepting illegality and the vulnerabilities which come with it. I have argued in Section 7.7 how the relative autonomy of the hacker scene have been historically and socially constructed. I have looked at several factors including the political economy of IT work, the decades long process of institutionalisation in the hacker scene, the social conditions necessary for the formation of an alternative engineering culture and others. These factors are deeply intertwined with the social history of late capitalism and explain why this particular social group can and do maintain their own social clubs and shared laboratories. Of course almost all social groups have their specific social spaces, but few are systematically peer produced and self-managed as the hackerspaces. Many use technology to organise their social life but few design, debate and produce custom technologies that are geared towards realising their specific forms of sociality in the face of social conditions that are in many ways averse to it. Door systems are such technology, tied to the form of life that belongs to the hackerspace. Hackers adapt to changing social conditions through grassroots innovation.

Coordination of the members' movements is a hard problem for several reasons. There is a great diversity of schedules amongst the membership: some work exclusively during the day and some during the night. The degree of flexibility and spontaneity is also different: some have regular hours while other have a hectic life style. Finally, different members want to do different things in the hackerspace, from working alone or with a few friends in deep concentration on a single project, through co-working with a bunch of others with some breaks when crossovers happen, to throwing a party where all the technology enthusiasts of the city could attend. While the most wide spread use case for door systems is to bring people together, they are also used to keep people apart sometimes. During my field work I often heard from a member that she is happy to go to the space if somebody else already opened it, or that she goes to open it hoping that other people will join – but on a few occasions members said that they prefer to go in quiet moments when they can be practically alone with the machines in the hackerspace. There are door system implementations which offer readings of the number of people in the hackerspace, or even some of the individuals' handles. These allow members to find or avoid each other on an individual basis without too much hassle.

Media like the door system allows for a space-time as usual, so that members can know about what is going on at a distance, and they can adjust their movement to the space state in real time. As pharmakons, these features are obviously the cause as well as the proposed remedy for the era of multi-located work which disperses (potential) communities and divides the workers across different spaces. They also address the specific needs of precarious workers, independent consultants and freelancers who cannot afford to rent an office of their own. The nonstop access policy suits the distribution of work across time too. On the one hand, since these types of workers rarely have a continuous employment. On the other hand, when they do have a work it often comes with tight deadlines that compels them to do all nighters. Of course, the advantages of the door system and therefore the hackerspace itself quickly evaporate if one is not an inner city dwelling knowledge worker who can easily go and work from a laboratory.

Finally, the door system as an unfinished artefact that enables coming together in an unfinished architecture addresses the problem of individuation that is the key characteristic of reflexive modernity. Individuation separates workers to individuals who are required to give a *virtuoso performance* in the words of Virno (2004). The hackerspace does provide a space for the expression of individuality and for the display and development of virtuoso performances, but at the same time peer production practices allow for a collective experience that thrives on those. The door system is usually restricted to signalling open or closed and does not go into detail about who did the opening or closing. As a result, members often go to the space for the general experience of meeting the community of members rather than this or that specific person. The same can be said about the experience of building the door system, which at least over the years becomes a kind of collective enterprise for members (Section 10.4). Similarly to the r0ket which calls forth a people (9.5), the ritualistic aspects of the door system forge the community together (Section 10.2). When one opens the hackerspace there is a sense of a duty performed as a virtue in the service of the public and in the name of a specific community.

The media that hackers prefer to document and develop the door systems and coordinate the other matters of the hackerspace are also supportive of the tendency of collectivisation. Mailing lists, wikis and IRC channels share a common characteristic as opposed to user profiled on social media monopolies such as Twitter and Facebook – or even traditional websites. Users can address their questions and comments the collective of the community as a whole by posting a message to the mailing list, editing a page on the wiki or asking a question on the chat channel of the hackerspace. They can choose their own pseudonym (there is no real name policy) and speak as an individual to a collective. In turn, members who read the communication can also answer as individuals without stepping out of the formal context of collective, because their enunciation is framed by the collective medium. Compare this mechanism to social media monopolies where user profiles are either individual profiles of discrete persons or collective profiles of organisations. Even though what I described above sounds rather straightforward, on the latter platforms one has to choose between addressing a person as a person or a cohesive as a collective. Those media are structured by the logic of representation where collectives subsume individuals, whereas on mailing lists, wikis and IRC channels individuals can choose to address collectives without forcing people on the other side of the line to assume that collective identity to answer. Once again, both door systems and the preferred communication platforms of hackers represent the cutting edge of individuation at the same time as they also offer a practical critique, allowing people to band together without being confined to a single identity.

Constructing door systems usually involves components purchased from the market as well as recycled components, in the same way that they involve traditional engineering techniques as much as technical tricks that only hackers would think of. While most hackers have to work to earn a living and make money for pet projects, their privileged position allows them enough freedom to decide where to invest a significant portion of their time and money. Furthermore, due to their expertise, they are able to valorise limited resources to greater effect than some other social groups and organisations. I have shown in Chapter 6 that recycling hardware have been instrumental in the foundation and day-to-day operations of hacklabs and it is still a backbone of running a hackerspace. Of course FLOSS practices mean that much of the software programmers produce in and outside of the world of work can be freely used, studied, changed and distributed in hackerspaces. The latter example shows how relative autonomy does not mean isolation from the state or the market, but a dialogue at a distance where hackers can formulate their own concepts of technology which can in turn influence market practices and employment conditions. Thus relative autonomy enables the establishment of techno-social spaces which can accommodate a wide number of uses and a variety of participants with differing backgrounds, though the claims of universality they make remains more of a vision than a reality, as the gender example exposes.[147] In essence, the primary product of hackerspaces is a specific form of self-organised and technologically mediated (male chauvinistic) network sociality.

10.6.6 The project order

Like network enterprises, hackerspaces are built around the idea of *projects*. It is quite astounding how the *project order* described by Boltanski and Chiapello (2005) is hard wired into the way hackerspaces are organised, and how consistent

[147] Not to mention that racial and other bias have scarcely been explored in the hacker scene to the extent that gender has, neither in the discussions of participants, nor in academic literature.

this order is across various hackerspaces. Work is traditionally coordinated through the combination of the following information and communication technologies: a wiki, mailing list and chat room (from least to most ephemeral). The most prominent section of the wiki is always a *projects* area with an overview of who is working on what, as well as the stage and progress of the project. Projects are usually as concrete as "build an OpenPilot drone", but they can be abstract too, like "motivate people to clean more often". I already argued that the expression of technological creativity through the production of small scale electronic artefacts is but a byproduct of the specific form of sociality associated with hackerspaces. To engage with the social dynamics of the space or to create a material artefact are conceptualised on the same level by practitioners. Door systems are good for showing such univocity because they explicitly involve both things, engaging with how people use the space through a technological intervention.

In the spirit of open collaboration which is the hallmark of peer production, project development is undertaken through public documentation. That is why public documentation is as essential for running the organisation as an unfinished architecture as it is essential for making small scale electronics (unfinished artefacts). The information available to the general public on the website is often the same information which members use as the internal documentation for the everyday work on the prototype. This is complemented by the *open door policy* of the space: anybody can walk through the door and contribute to the initiative. Interestingly, the initial account of technologically mediated mass online collaboration given by Shirky (2008) also singles out the wiki as the tool of choice for supporting such a social dynamic. Technically, the wiki engine software used by hackerspaces is often the same which powers Wikipedia – one of the prime examples used by theoreticians of peer production.

Door systems are created with the explicit intention to allow for a specific kind of sociality to work in the hackerspace, but their construction is not purely instrumental. Since hackers like hacking and they often go to the hackerspace to work on OSHW for a change, it is a compelling idea to confront the social problems presented by the form of life members have to live under late capitalism through technological means. These allow them to collaborate with their peers while improving their knowledge and gaining experience. Peer production practices of open collaboration are enabling such activities, but they are not simply an anticapitalist gesture. Rather, they rearticulate the project order that is hegemonic in contemporary capitalism. The project order as it is implemented in the construction of door systems varies from the model described by the French sociologists in a number of ways.

First, goals are self-directed, being offered by the environment as much as emerging from the individual interests of members. A member with interest in electronics may look around for something to do with their new Arduino or to try out some ideas for home automation using the zwave protocol that is geared for such use. In this case the interest is in exploring the technology itself rather than achieving a specific goal: what could be called the instrumental use of technology. It just so happens that a recurring problem is that people leave a window open when they close the hackerspace, so that it would be nice to install a sensor that warns them when they toggle the space state.

Second, processes are artefact-driven rather than structured by project leaders. Continuing the interpretation of the previous case, it could happen that another member becomes familiar with the window warning sensor and picks up the idea, developing it into a contraption that actually closes the window automatically instead of merely sending an alarm signal. While projects are tracked and directed by managers in the corporate environment, in the hackerspace there are few people who are interested in directing the efforts of members, while many members track what is going on in a daily basis. The composition of project members can thus change dramatically. The information about the project is theoretically passed from one person to the next through the documentation, but in practice the actual artefact plays a much more significant role. The latter effect is enabled by the specific functionalities, design principles and construction methods of unfinished artefacts which make them easy to understand, modify and deploy.

Third, projects organised according to the project order start by enrolling a heterogeneous troupe of participants, and then run their course just to come arrive to a clearly set goal and dissolve the associations, which causes precarious working conditions and forced flexibility of labour. However, membership in hackerspaces is not tied to projects. Indeed, some members never really do any project in the hackerspace, while most get involved in a dozen or half projects during their membership, and some join explicitly to carry out specific projects. Importantly, projects in the hackerspaces do not necessarily come to a conclusion: that is one of the main reason that I decided to call them unfinished artefacts, based on the derogative comments of makers who used such a fact to distinguish themselves from hackers. However, projects are evaluated according to a multiplicity of criteria, only some of which reference the finished artefact.

To conclude, the project order is rearticulated in hackerspaces to be self-directed, artefact-driven and process-centred. These factors allow door system implementations to persist in time as material infrastructures sustaining the social dynamic of the hackerspace while growing step by step over the years. However, they also allow the initial setup to persist for a number of years without any modification as long as it proves to be stable enough to endure continued use and offer enough features to satisfy the needs of participants. As mentioned before, unfinished artefacts are not stabilised much so that they can always be opened up and elaborated. The rearticulated project order becomes another component in the unfinished socio-technical architectures of hackerspaces.

10.6.7 Cognitive capitalism

Conceptually, hackerspaces are at the forefront of transformations in contemporary cognitive capitalism. As firms are present on the market through their products, hackerspaces participate in the scene through their projects. This is reminiscent of what Arvidsson and Peitersen (2013) termed as the moral economy, where a multiplicity of values circulate, yet turning the social capital of reputation to financial capital is structurally easier. While the product is characterised by having an exchange value which enables it to circulate on the market, the project circulates in knowledge networks without exchange value (see the arguments in Section 9.2). Contemporary mass self-communication networks outperform the market as distribution channels, so that projects can spread faster and further then products. Partly because of these factors in off-the-shelf political economy, but partly because of the political economy of their cultural valuation (orders of worth), they have more freedom than products in the sense of what can happen to them and what kind of meanings they can take on (see Section 9.5 for an overview of meanings attached to the r0ket device). I am convinced that the fact that the projects of hackerspaces are essentially *counter-productive* in the sense of lacking an exchange value is still under-appreciated. This observation has to be considered against the universal background of all artefacts – human created objects – in the world. As I argue in Section 9.2.5, the product is the universal form of artefacts in capitalism, especially small scale electronic artefacts: in fact they are called “consumer electronics”. It takes much culture for these open hardware experiments to escape market circulation. Of course *successful* projects escape the ghetto in the way music and musicians, styles and celebrities make their way to the limelight. They are picked up by entrepreneurs and transformed into mainstream products. But the vast majority are not successful in the commercial sense, nor aim to be. Moreover, as the story of MakerBot in Section 7.1 shows, commercialisation is not a straightforward or unconflictual process.

Projects often have built-in functional elements which seek to counter commercialisation Aibar and Maxigas (2014b). The most important of these is not the licence but the documentation. Whatever the licence, the documentation demonstrates prior art and therefore it can potentially be used to attack patent applications. Hardware solutions cannot be licensed in the same way as software, so that patents are the main legal instrument of ensuring privileged access to their production. Software patents are recognised in the United States but not in Europe. In fact the European directive on software patents was dropped in response to massively distributed public protests in which the hackerspaces and their members took an active role. These differences notwithstanding, the free software commandment of “publish early, publish often” is not simply a commitment to the community but also the first step against commercialisation. The project order practice of using a wiki to develop and simultaneously document projects has to be understood in such a context.

Culturally, the dividing line between products and projects is even more subtle.

Projects usually manifest themselves as small scale electronic artefacts which one can hold in hand, and use for a more or less recognised functional purpose. However, they are usually evaluated as a process rather than as a product – one of the reasons I prefer to call them *unfinished artefacts*. Such evaluation points beyond what is traditionally understood as use value. A small catalogue is sufficient to demonstrate these criteria. A good project uses cheap off-the-shelf parts. It sports a simple yet elegant design which is easy to understand. The plastic box which became the hallmark of consumer electronics is often missing, laying parts bare to the eye of the beholder. These factors lower the barrier for *reproducability* which is fundamental to the value of the project. Alternatively, off-the-shelf parts can be replaced by components "found" in the environment, so that the project presents a compelling case of restoring their use value. Furthermore, a good project is not a routine undertaking but enables its maker to move out of her comfort zone to learn new skills and gain new insights. For this reason it does not have to be innovative. Reinventing the wheel without prescience of prior art is considered a mistake by hackers, but imitation is totally legitimate. These are evidently *educational* considerations. Due to limitations imposed by the environment, the imitation is adapted to the specific circumstances anyway, so that its value comes from how it sits in the local techno-social context.

A project can also excel in *engineering aesthetics*: surprisingly clever solutions and strikingly elegant design decisions make a hack in the highest sense. Hackers appreciate artefacts which achieve complex behaviours using simple rules, thus connecting engineering aesthetics to reproducability. Furthermore, the self-recognition of hackers in the hacks elevates the *cultural value* of the project. As with any subculture, certain signs have a special affective and cognitive meaning which is greatly appreciated. Putting aesthetics and self-recognition together, Coleman (2012) rightly argues that in-jokes are the heart and soul of the hacker scene. Last but not least, there is what is conventionally called use value, which determines how useful the given artefact is as a *practical tool*.

Ultimately, however, the simple fact that somebody wants to do them is enough justification and legitimation to work on a project in the hackerspace – projects are not formally evaluated. In sum, as I have shown in Section 7.3, hackerspace projects combine three functions that are otherwise strictly separated by the modern institutional grid: *education, research and production*. What needs to be grasped here is that the value of a project – the *orders of worth* through which it is evaluated – cut across categories and these categories themselves are often circumstantial to the abstract use value or the sometimes missing exchange value of the actual artefact. While the market creates its own context so that products can be judged independently of the process that spawned them, unfinished artefact can only be understood and therefore valued in context.

These make them stand out a lot against the background of old capitalism. However, if we take into consideration the thesis of cognitive capitalism that analyses contemporary transformations, then it seems that unfinished artefacts seamlessly blend into the picture. Just as the main source of exchange value is produced by market externalities in cognitive capitalism through models of capital accumulation that were not centre to previous editions of capitalism, the use value unfinished artefacts is largely circumstantial in the sense that their significance is far more varied than being instrumental tools. Therefore, the revolutionary character of unfinished artefacts should be seen in a critical light, knowing that they are not necessarily an unintended consequence of technological progress but a structural effect that reflects the hard core of capitalist tendencies. Nonetheless, the difference between mainstream models of capital accumulation in cognitive capitalism and the production of unfinished artefacts like door systems in the hackerspaces is that the former is still oriented towards exchange value even if through indirect loops, while the latter is squarely rooted in unalienated labour: an experience that points beyond capitalism. While such an experience incites participants to set up organisations for its cultivation, Rundle (2015) is right that these do not constitute a social force in their own right and for the moment it is hard to see how they can lead to structural transformation.

As with the project order, it is fair to state that hackerspaces rearticulate the dynamics of cognitive capitalism and the network enterprise. While there are significant differences between versions of cognitive capitalist business models of capital accumulation and the self-managed techno-creative self-expression of hackers, cognitive capitalism is a concept of a totality. Consequently, subversion could technically not come from the outside. Instead, the challenge to cognitive capitalism would have to arrive from the development of its own contradictions – a process that is surely moved forward by exploring peer production practice for the construction of small scale electronic artefacts, and may even account for the ambiguous political position of hackerspaces.

Looking at door systems, they formally conform to leading cognitive capitalist models of accumulation and capture. First, it valorises everyday gestures and produces an environment that is immanently productive. Toggling the space state becomes part of a routine that contributes to the everyday running of the hackerspace as well as serving to signal its marked moments such as becoming a hackerspace or showing the hackerspace to newcomers. Pushing the button is not much of a task, yet it adds value to the hackerspace which is a productive activity. Second, most door systems turn these everyday gestures into metadata that is logged, mined and visualised. The SpaceAPI as a standard interface makes this relatively easy even for third parties, so that the value does not necessarily materialise on the side of its producers, and it may or may not be shared in a common pool accessible for everybody. Hackerspaces often keep tabs on their space state though and save its history for analysis. The logs are transformed into graphs and displayed on the website as a proof of activity. Characteristically for cognitive capitalism, the very abstract information that somebody was in the laboratory is enough to contribute to the creation of new social and technical connections which are usually beneficial for the hackerspace. Third, the overall effect is a form of social control which works through surveillance and nudging rather than discipline and punishment. The visualisation of statistics motivates hackers to keep the space open for more time as well as to use it for projects and social gatherings, otherwise it becomes clear that it is not worth to pay the rent, or at least participation could decline to levels below sustainability. The physical interface provides feedback often in a theatrical way which compels members not to forget their duties of operating the machine. These measures are also suggestive of the epochal shift that Deleuze formulates in a shift from Foucauldian disciplinary societies to what he calls the societies of control (Deleuze 1992).

Google as a prototypical cognitive capitalist corporation provides a free service in exchange for enrolling users in their system of accumulation and capture. The Gmail interface turns the everyday gestures of communication between friends, family and business partners productive of a database about users. It structures the interaction of people in a particular way that encourages them to produce more data about themselves. The backend infrastructure transforms user data into a valorisable advertising database at the same time as it serves as a platform for the display of the same advertisements. Once again, the crucial difference is not the systems of exploitation but in the political economy of ownership, management and maintenance of the system, as well as the way it is designed to distribute participation and expertise. It is easy to see that despite its similarities to Google's Gmail interface, the door system is designed, built and operated mostly by its users, and they are the ones who benefit from it the most, while also contributing to a knowledge commons through door system designs and space state data.

In the larger scheme of things the hackers' version of cognitive capitalism and the hackerspace as a network enterprise may not differ as much from their mainstream versions as some would like to think. However, it may be possible to fill the same social forms with different content. Social conflict may materialise even if social forms are remarkably symmetric on the two sides of the barricade.

Hackerspaces are owned and run by members for the good of the general public, while network enterprises are owned by capitalists and run by workers for the shareholders. Similarly, projects are supposed produce exchange value while projects as their are understood and practices in hackerspaces are essentially counter-productive: that is, at best they produce use value. The emic term *pet project* is a wonderful name for unalienated labour because cultivating pets is also something that one does outside of work just because it is personally and perhaps ethically rewarding.

10.6.8 Conclusions

How participation in a flexible working space of technology enthusiasts is organised through conventions and technologies in the context of large scale social transformations? I tried to connect the birds-eye social history of capitalism with the organisational sociology and ethnography of hackerspaces, and these two to the post-digital archaeology of technological artefacts. I followed three methodological steps, sometimes repeated recursively.

First, I sketched out my interpretation of the transition from the modern phase of capitalism to network and knowledge society, and how it transforms the daily realities of workers. It has been established that both information and communication technologies (network architectures) and the workers who operate them play a central, privileged and in some sense hegemonic role in the contemporary configurations of capitalism. Some technology workers exploit these possibilities to construct areas of relative autonomy such as the segment of hacker culture that are the hackerspaces.

Next, I repeated the claims mostly developed in Chapter 6 that the existential conditions of hackerspaces are tied to the specific employment structures of late capitalism as well as that hackerspaces express the contradictions of late capitalism. In an attempt to escape social determinism I also showed how the hackerspaces established an interactive, sometimes subversive, sometimes symbiotic relationship to their political-economical environment. In a more or less arbitrary move, I singled out the line dividing the hacker life and the professional life of individuals to serve as the interface between the hackerspaces as an organisation, and the precarious-flexible labour market. The main line of argument here was that various groups of precarious and flexible workers[148] band together in the hackerspaces for various reasons, but all in a reaction to their positioning in these matrix of transformations.

In the final move I showed how the expression of technological creativity through the creation of small scale electronic artefacts which is the key characteristic of shared machine shops shapes hackerspaces as a scene and as an organisation, tracing out a shift in the regulation and coordination of workers. While the hackerspace itself can provide a shared space to meet, the door system has to be there to provide the shared time. The door system is an invention of necessity for maintaining a particular sociality between workers who are isolated by changing social conditions.

In summary, technology workers negotiate the establishment of a shared time and space by using their expertise to incite participation. Hackerspaces are not significant sites of political organisation, yet they are important for their users for preserving the community practices amongst technology workers that allow them to be in solidarity with each other. Using the results from the previous sections we can assert that hackerspaces are interesting because their members think about social relations directly in matter, designing material infrastructures around desired social scripts without the mediation of either the law or the market.

[148] Whose precariousness and flexibility can significantly and independently differ from each other.

10.7 Shadow: door systems versus time clocks

Drawing the historical perspective even wider, the significance of door systems can be assessed by vetting them against the time clock – a machine to measure work time in an industrial setting – that is as old as industrial capitalism itself. Here the principal source is renowned social historian E.P. Thompson (1967). Peasant farming, craftsmanship and the cottage industry were forms of work where workers had almost complete control of the work process that they had to adopt to the rhythm of the crops or the seasonal demands of the market. These types of work are described by Thompson as task-oriented labour. He cites a wide range of pamphlets, letters and regulations which sought to combat on moral grounds what capitalists saw as the deeply rooted idleness of the poor. These documents give a clear idea about the kind of morality that was soon inscribed in the mechanical contraptions called time clocks. He emphasises that the logic of the time clock – e.g. the regularity of work, the synchronisation of production and punctuality – were simultaneously introduced in the other spheres of life, most importantly the school that was destined to introduce children to the "habit of industry" (1967, 84).

The first recorded system of check-in were introduced by a capitalist called Wedgwood around 1780, while the first actual machines that printed the entry time of workers were manufactured by Harlow Bundy in the United States in 1885. The contraption proved to be so successful with factory workers that he soon patented the item and formed the Bundy Manufacturing Company. The company merged with others to form International Time Recording Company in 1900 and on further mergers changed its name to International Business Machines.

IBM went down in history as the market leader in calculators and later mainframes and personal computers. Levy (1984) explains how hackers saw IBM as their arch enemy in their struggle for the personal computer, since its routine was to give control of the computer to operators who would take punch cards from programmers, thus forming an anointed clergy that separated users from their tools. However, with the success of the personal computer IBM changed its ways and became the manufacturer of accessible and modular devices for every household. Later it came to embrace FLOSS in its business strategy as the first major corporation to do so, contributing considerable financial support and work time to Linux kernel development. Partly for these reasons its line of laptops marketed under the ThinkPad label became de facto standard in hacker circles. So much so that in contemporary hackerspaces members who launch tirades against artists who cannot use Linux or do microcontroller programming with anything other than Arduinos refer to these inferior class of users as "Mac users". Apple computers came to epitomise the opposite end of the spectrum as IBM mainframes of old times: while in the case of mainframes it was the clergy who separated the users from the godhead, in the case of Macintosh computers it is Maya's veil, the User Interface, which keep users in ignorance. Therefore the history of the time clock is itself in a complicated relationship with the evolution of the hacker scene.

In a very Latourian way the word timekeeper can refer to a machine as much as a job fulfilled by a human. In fact in early factories timekeepers were employed to keep track of work time, before time clocks were introduced. Just like the butler who closes the door until its job is overtaken by a dedicated machine as discussed in Section 10.5, the time clock is none else then the inscription of a social function into a material form that is more resistant to change in general and more easy to discipline in particular. Around 1700 a Monitor was a person who manually recorded the time workers spent in front of their machines – which in turn actually required him to be in control of a machine: the clock or watch. Around the same time workers complained that they did not have watches, or ones that had watches got them confiscated by factory owners, so the control of the time was the exclusive power of the capitalist. Early IBM time clocks were described as

> automatically stamp on a card inserted in a slot in the clock by the workman the time of his arrival and of his departure (Gillette and Dana 1909, 110–111).

The shift from Fordist to post-Fordist capitalism of course changed the context in which time clocks were deployed. The eight hour working day that was in a way the pinnacle of the class compromise effected by the classical workers' movements eroded in many areas of work thanks to a historical restructuration. On the one hand, jobs have moved to the periphery of global capitalism where the power relations between the working class and capitalists were configured differently. On the other hand, many workers in the core economies came to perceive regular work hours as detrimental and fought for their freedom demanding the flexibilisation of work. As a result, in the core countries knowledge workers nowadays are either doing piece work as freelancers or getting paid by their employers after the net hours put into a project. The social function of time clocks changed accordingly to count the number of hours spent on a project at times that are convenient to the worker rather than enforcing a working day of a fixed number of hours. Since the number of hours are largely self-reported in the contemporary time clocks, the system shifted from the discipline and punish model of first modernity to the surveillance and control model of second modernity.

Of course this does not mean that Fordist workers were not hacking time clocks. No instrument of work discipline is introduced without substantial resistance from the working class. Since each worker had their own punch cards, or at least their individual numbers to put into the time clock when they passed the machine at the perimeter of the work place, by far the most common method of sabotaging the time clock was punching other workers' cards or numbers for them. Such methods have been especially wide spread in the socialist economies of full employment where there were more workers than work but the physical presence of workers was nonetheless expected by the masters of industry. Indeed, workers evidently understood the role of time keeping in the factory regime since the beginning. They specifically targeted clocks during revolts, putting them out of order.

Time clocks also stood at a crucial transformation in the conception of time when it became equated with money. Time clocks were the material instruments where such a social imagery became implemented into an impersonal, mechanical, material instrument which enforced it through manipulating the rules of nature – as contemporary engineers thought about their machines. Since proletarians sold their labour power by the hour, the time clock measured how much they sold and thus how much money they would get for their work. Indeed, such system would be actually an improvement from piece work, which constituted an earlier and lower category of jobs. Piece work jobs were payed after the number of products turned out, in the style of the cottage industry regime.

Fixed monthly salaries, and the obsolescence of time clocks was characteristic of what Frayssé (2013) calls "Fordism 2": a middle period whose guiding paradigm was regulation instead of efficiency, or better, the regulation of efficiency. Resistance of the working class resulted in a historical compromise known today as the model of the welfare state. However, this was partly financed through debt, and became increasingly infeasible economically, while at the same time its political raison d'être eroded. The 1970s marked the beginning of the restructuration of the production regime. Frayssé largely agrees with Boltanski and Thévenot (2006) that the ethos embedded in these transformations can be formulated as "*freedom*", even though the latter uses the leftist phrasing instead of the liberal one: "*autonomy*". One could argue that the third – post-Fordist – phase depends on the self-regulation of efficiency, where the limits of capitalism are internal and work discipline is internalised. Ironically, in terms of work discipline established through the regulation of time, this period marks a partial return to piece work and the cottage industry, with the work of self-programmable independent consultants on one end of the spectrum and the disposable general labour of Amazon Turks (Gray 2013) at the other. As mentioned before, those whose work is not regulated by piece is still regulated through updated versions of the time clock – some of which are still supplied by IBM in the form of computer software.

Of course phases of capitalism do not merely follow each other. Since it is a historical process, at least in terms of the work regime there is stratification happening. Namely, basic methods for the establishment of work discipline remain to be found, but they are distributed according to the level of privilege different groups of workers achieve. In contemporary capitalism, the highest strata of workers enjoy a stable monthly salary, the middle range are payed by the hour, and the cheapest freelancers in the periphery work for a piece rate. The last is the ideal for the organisation of projects, while the first is the best for making a living. In sum, the social conflicts around the idea of workers' control over the production process can be formulated thus: is it the workers conforming to work, or work conforming to workers?

One could argue that hackerspaces are spaces of leisure and thus they have nothing to do with work in general and work discipline in particular. Therefore, it is impossible to discuss time clocks together with door systems on the same page. Even though such an argument would disregard much of the discussions, I feel that it has to be addressed and refuted here.

First, post-Fordist theorists aligned with the late Autonomia thinker Mario Tronti write about the social factory: they mean that the locus of production is more dispersed in social life at the end of the twentieth century than in the beginning (Thoburn 2003). In Section 10.6.1 I reviewed some similar arguments from the sociology of labour. Previously I emphasised the mechanics of networked labour and the project order, pointing at hackerspaces as privileged sites for cultivating a strong network of professional peer who can be part of any paying project if there is a need for their participation and expertise. On the one hand, as networks need to expand indefinitely, random visitors to the hackerspace are not an obstruction. On the other hand, since project members change from project to project, it is good to have a more or less stable group of collaborators that does know how to work with because of previous hackerspace-based non–paying collaborations.

Second, hackerspaces play a role in the innovation ecosystem that eventually leads to products. They serve as a protected niche of experimentation out of the reach of management and market pressure as well as institutional agendas. Therefore, even though there are few products developed in the hackerspaces, the technological work in the hackerspaces can lead to product development. As documented in the first section of this chapter, even the research into door systems – which are very specific to the hackerspaces milieu – turned up cases of commercialisation of unfinished artefacts, while the r0ket that was never meant to be sold has been acquired by Pollin in at least 300 copies. The MakerBot story recounted in 7.1 provides a high profile case of a spin-off company started from the research results achieved collaboratively using the material infrastructure of hackerspaces.

Third, as one hackerspace member reminded me, participating in pet projects and conversing with people with expertise in a wide range of areas is actually essential for technology professionals. On the one hand, technology itself is moving fast and getting more complicated, so that large technological infrastructures have many layers that each present their own problems. On the other hand, each position in the industry requires a very specific skill set, so that professionals cannot afford to be stuck in their niche if they hope to find a next job in the technology sector. The corporate environment is most often not very conductive of gathering a wide range of experience, and that is where pet projects and socialisation can help. ICT workers use these subjective and objective opportunities to follow numerous specific personal interests. Any of these personal interests can turn into a professional responsibility in their next job, or may come in handy when facing a complex problem that involves the interaction of several layers in the technology stack. Furthermore, it is a recurring pattern in FLOSS and OSHW development that companies hire people who worked on side projects in their free time and proved to be good at what they did.

Fourth, unalienated labour is still work and needs to be organised somehow. The door systems are the way in which it is organised in time through material artefacts, in contrast to the classic Fordist factory where time is organised through material artefacts like the time clock. Even if unalienated labour is spontaneous, it needs discipline to bring it to fruition. Moreover, door systems are not mere tools for coordination: they also *keep time* in the sense of writing logs and drawing up graphs of the activity of labourers.

Fifth, even if all the above is false, coordinating meetings in the hackerspace is still something that members have to do. Most members have to juggle their professional responsibilities with hackerspace life and other areas of their life, and even if they have quite a bit of flexibility compared to other social groups, their official responsibilities at their paid work are oftentimes the least negotiable. Therefore, even if they would simple negotiate their labour time with their other activities and with their peers, they would still have to do it *as workers*. Incidentally, these reasons also explain why it is generally a mistake to discuss hackers of any kind strictly as hobbyists, even if leisure in general and the specific social practices associated with hobbies can partially account for the dynamics of engineering subcultures, as we have seen in 8.6 during the discussion of Men's Sheds. A good summary of these points would be that in the context of the hackerspaces there is no fun without profit, and no profit without fun – there is no escape from the social factory.

The classic capitalist worker wakes up to the sound of an alarm clock and hurries into the factory to punch her card with the time clock. Such a regime of control brings together bodies of workers in a predictable fashion in order to facilitate production. Meanwhile, in contemporary cognitive capitalism, the somewhat subversive but still very timely model of the hackerspaces works in the inverse direction. Productive individuals do not have to be woken up at the same time by machines (they will work driven by their passions all night anyway): they coordinate in real time through the door system. When eventually one finds its way to the hackerspace, she flips the switch or pushes the button and the call for work goes out through the networks of mass self-communication, from chat rooms to Twitter. These signals gather the members of the hackerspace. In the final analysis, it is not the clock which controls the worker: it is the worker who controls the clock! Such regime is not predictable any more, but flexible – which enables productivity to be optimised on a fluid biopolitical level rather than on a spatio-temporal grid. Workers work when they are in the best condition to do so, and in a manner which maximises their cooperative potentials.

Curiously, such phenomena is consistent with Thompson's description of task-oriented labour. Once, task oriented labour is more humanly comprehensible than times labour, since the task at hand and the time that it is good to work on it is perceived as a necessity – consistent with the dynamics of pet projects. Twice, it shows little demarcation between work and life, so that social intercourse and labour are intermingled – consistently with my results about peer production practices in the hackerspaces. Three times, it appears to be wasteful and lacking in urgency according to men accustomed to labour timed by the clock – consistent with maker's complaints about hackers that they are never finished with their projects. Interestingly, these criteria also describe what I called the artefact-driven work process where participants of the project can completely change without endangering the completion or continued development of the project. Indeed, Thompson notes that such "work pattern was one of alternative bouts of intense labour and of idleness, wherever men were in control of their own working lives" and notes that task-oriented labour pattern "persists among some self-employed" to his day (1967, 73).

In conclusion, the problem and the solution to the time clock problem have both changed historically. On the one hand, in the hackerspace as an unfinished architecture working according to the principles of peer production, tasks are self-assigned and therefore time is self-managed rather spontaneously. In contrast to civil society organisations where volunteers take on responsibilities before hand that often have pre-assigned time-slots associated with them, in a hackerspace the members work on projects as much as they want and whenever they want. I have pointed out that this difference is due to the fact that hackerspaces do not provide a service to the general public, or at least not in the way it is usually conceived. On the other hand, the door system as an unfinished artefact – a small scale electronic artefacts which is both peer produced and supporting peer production practices – have been designed, built and it is maintained by members for the good of members and the general public. Therefore, its operation is both transparent and open to changes, improvements or even sabotage by members. While a traditional time clock have to be tamper proof – in some early factories clocks were guarded by a dedicated time keeper person as well as physical chains – its reliability is dependent on the trust between members. Similarly, the work discipline infused by a door system is not imposed by discipline and punishment but by surveillance and control – the broadcasting and sharing of information publicly allows for a certain peer pressure that helps to keep members staffing the hackerspace, and bring them together to form a community. Once again, the main difference between a time clock and a door system is that a time clock is the capitalists' instrument to impose a morality on factory workers, while the door system is the members' contraption which is itself a manifestation of their shared values as much in terms of engineering as in the manner of organising productive labour.

10.8 Conclusions of the second case study

Door systems have most important properties of hackerspaces inscribed in them – therefore understanding door systems is understand hackerspaces. Door systems are spread out in a tangible architectural space while they also have their digital counterpart and none works without the other. They are also open for participation in their design, use and development. They serve as an infrastructure which is networked in various ways to form the large scale material infrastructure of the hackerspaces as a scene. Just like hackerspaces as particular organisational forms, they proliferate based on people grasping the basic idea in a particular geographical location and then implementing it according to the local context elsewhere.

Such a setup enables building a scene which is coherent in terms of its technological infrastructures and shared social norms but at the same time responsive to the local human geography. Hence people can develop ideas and contribute creative solutions to the whole scene which are different from everywhere else without holing up in an ivory tower which would isolate them from their peers. I demonstrated these tenets through a survey of door systems installations in five different locations, even though the field research for the case study span over three dozen hackerspaces and each had their own version of the door system. Internally, when hackers develop their door systems they are hacking at the actual architectural properties of their shelter as much as on the social architecture of their organisation - which makes the door system a veritable "architecture of participation" (Shirky 2008, 17). In this vein the conclusion came that hackerspaces combine organisational innovation with technological innovation.

However, such a notion have to be qualified, since innovation itself happens in innovation networks where hackerspaces can play but a small part. Indeed, as the door system eminently shows, the significance of hackerspace for innovation is exactly their isolation from the evils of state regulation, market pressure and competing institutional agendas of the academia and civil society. The peer production of unfinished artefacts is described as unalienated labour and as such an expression of undirected creativity because the motivation can be as much as boredom or inconvenience. Once again, working on the door system can be much more or little less than manufacturing in the sense of making something. On the one hand it can be research as in the case of Bitlair members reverse engineering the security system that came with their barn only to end up springing the secret service of the country to action and forcing the vendor to issue a new an improved product line. On the other hand it can be education as when ideas from other hackerspaces are reproduced with the help of their designers, so that local members learn something that is new for them. These results support the ongoing thesis that hackerspaces mix research, education and production in a seamless way that challenges the compartmentalisation of human activities within the modern institutional grid.

The close connection between hackerspaces is shown in their ritualistic significance. Door systems mark boundaries immaterial, imaginary and symbolic as well as material online and offline in many different ways. From a hackerspace integrating into the material infrastructure of the scene by implementing a door system compatible with the the SpaceAPI, through starting to function as an actually useful hackerspace in the moment the space state is toggled, to a new member getting the keys of the space and right to open and close by handling the physical interface to the door system, these small scale electronic artefacts play an indispensable part in the unique engineering culture of hackerspaces. Door systems structure participation in the hackerspace that answers to the universal aspirations of the scene as well as its sensitivity to the meeting with the unknown. The figure of the strange in terms of non-human phenomena and the stranger in terms of human phenomena is inscribed in the unfinished architectures of the hackerspace as an organisation through the functionality of door systems.

Door systems could be an example of the distributed manufacturing of OSHW because they are reproduced in a high number of hackerspaces simultaneously and their creators recognise, understand and develop implementations found elsewhere. Except that door systems are not made following a shared blueprint which evolves with each implementation in the way that a RepRap family tree can be established. Door systems evolve in a rhizomatic way so that there is only family resemblance between them, but no lineage. In effect, door systems are reinvented and rebuilt each time somebody makes one – sometimes even when the same hackerspace moves to a new location. This is only possible in an environment where the general level of technical expertise allows local groups to reproduce results from similar contexts: therefore it is not only necessary to pass bills of parts and schematics around. Expertise have to spread together with the idea, and when that works, the actual bill of parts, schematics and the licence to reproduce them becomes rather irrelevant. The hackerspaces form a milieu where on the one hand nobody wants to reproduce exactly the same that others have done, and on the other hand inventions are usually documented and licenced but not necessarily documented well and licenced correctly. The result is that the licence is not a defining characteristic of these small scale electronic artefacts any more, even if they embody the ethos of peer production much more than most well documented and officially licenced products. Therefore, rather than calling them OSHW I call them unfinished artefacts.

Just as they are geographically specific, neither door systems nor hackerspace stand outside of history. The social dynamics expressed in and imposed by door systems, as well as the subjectivities they produce bear on themselves the stamp of late capitalism, and therefore express its contradictions. Hackerspaces are an example of self-management and collective organisation by one of the most privileged sections of the working class. Consequently, door systems are managed by their users and this shows on how they distribute agency through their technical functionality. In particular when compared with the time clocks where the tenets of capitalism have been inscribed since the rise of industrialism, they show distinctive traits of empowerment while they also answer to existential concerns raised by changing work patterns in late capitalism. Considered in the context of reflexive modernisation, they pose challenge to the institutions of first modernity such as the concept of a predefined and stable opening time, among other things. At the same time, as instruments of establishing work discipline under a cognitive capitalist regime they rely on surveillance and control through a feedback mechanism that runs through a public presentation of what is going on in the hackerspace. Furthermore, the mechanism of the door system and how it is documented and produced raises questions about the increased individualisation in second modernity. Door systems speaks in a unified voice for many humans through textual, visual and logical broadcasts, and thus it is a medium for the struggle for collective enunciation.

On a larger scale, the cooperation and coordination of the hackerspaces – partly through setting up diversely implemented yet functionally interoperable door systems – closely resembles the structure of the network enterprise. The role of the technology in constituting, maintaining and developing the hackerspaces scene as a networked infrastructure for hardware hacking and a social milieu for technology enthusiasts is rather unique. Naturally, all network institutions, be them states, network enterprises or manga fan clubs, have to build, maintain and develop material infrastructures through which they coordinate the projects they are working on and draw the lines of group memberships. However, state and market actors use formal, contractual links as the basis for such coordination, which the technical medium should follow – web designers call this the "business logic" to model in the application. On the other hand hackerspaces are reliant almost exclusively on the two things that are the most interesting to Science and Technology Studies scholars: material infrastructures and shared cultural norms.

11 Conclusions

11.1 From open source hardware to unfinished artefacts

The elementary research question I set out to answer was how peer production practices are transformed when they are adopted for the requirements of OSHW – particularly small scale electronic artefacts created through unalienated, self-directed labour in the hackerspaces. As Gershenfeld would say, can "almost anything" be peer produced? How peer production changes when we make this and not that? These are the basic questions belonging to the generality problem of peer production.

I started with the notion that the three organisational principles that Benkler gives in his definitive guide to commons based peer production do not necessarily stand for hardware (2006). **First, collaborating in *reliance on computers and the Internet* does not necessarily work because it is obviously hard to co-produce tangible artefacts telematically and especially because there is much tacit knowledge involved that is difficult to transmit in the form of digital signals.** Users may also lack access to the material infrastructures that are required for reproducing results. Hackerspaces address this problem by establishing tangible "architectures of participation" (Shirky 2008, 17) or infrastructures of peer production: actual buildings full of equipment and people with expertise that are open for collaboration. In many cases the simple fact that one can leave a project in the hackerspace and others can take it up – on the next day[149] or years later[150] – to continue the work, is in itself a great affordance that hackerspaces provide to their members and visitors.

Of course this does not mean that there are no computers involved. Hackerspaces rely on a standard set of digital tools geared towards the coordination of peer production activities: mailing lists, wikis and chat channels were three examples where members communicate and document their creations. I discussed an expanding taxonomy of similar organisations under the rubric of *shared machine shops*, and argued that they are typically caught up in agendas of more powerful institutions or prone to be subject to market pressure. Hackers rely on their privileged position as highly skilled knowledge workers to self-finance the operation of the hackerspace, so that they can focus on extending the possibilities for participation and cultivate their expertise in a protected environment. One indication is that projects do not have to be successful in any way: it is all right to fail in a hackerspace. This is where the idea of *unfinished artefacts* as eminent OSHW projects came from.

Second, the *lack of monetary compensation* does not necessarily work for hardware because there is always a cost of production and reproduction that has to be covered. Once again, hackers rely on their privileged

[149] As in the case of the r0ket badge.

[150] As in the case of many door systems.

social position and the relative autonomy they have built historically[151] to find their ways around these limitations. The r0ket team relied on sponsorships obtained from vendors and manufacturers, good connections to find the best deals and loans from the Chaos Computer Club to realise their projects, complementing volunteer labour. Door systems are routinely constructed from recycled parts and donations to the hackerspaces. Their status as infrastructure does not impose any deadlines so that they are never actually finished and can be elaborated as circumstances admit. Apart from these internal factors, there is also a sizeable niche market – partly operated by hackers themselves – that supplies electronic parts, microcontroller boards and single board computers which allow users to develop small scale electronic artefacts with relative ease. The mainstreaming of hacker culture widened the range of products that target this market segment, while the corresponding institutionalisation of the hacker scene developed interfaces through which the community can negotiate their interests with the industry and with regulators.

Third, the *lack of chain-of-command* can also be problematic because mistakes in hardware design can be complicated and costly to reverse while the risks involved can be higher even in the most elementary of cases. Indeed, just like in the case of complicated FLOSS projects, participants have to demonstrate a level of expertise before they are allowed to participate in technical decisions that have far reaching consequences. Decisions about the layout of the PCB (Printed Circuit Board) were made by a few people at the hard core of the r0ket team because mistakes – a few of which I discussed in detail – meant more work for everybody and endangered the viability of the project. However, a large number of volunteers could walk into the hackerspace and learn how to solder on the spot through helping out in fitting the m0dulbus connectors on the units that rolled down from the conveyor belt. Similarly, many door systems expose standard interfaces through which any member of the general public can access their resources or even active actuators in the building of the hackerspace. However, they also provide authenticated services like opening the actual lock on the door and functions such as toggling the space state that are the exclusive privilege of members. Therefore, the self-selection of tasks does not mean that everybody has automatic access to everything: limits to participation are set based on expertise proven through participation in the project itself. I went to great lengths to show how shared values in the hackerspaces are inscribed in the electronic artefacts (such as the r0ket badge) and the material infrastructure (such as the door systems) and certain behaviours encouraged or discouraged through scripts suggested through technological rhetorics.

Finally, in order to evaluate these aspects realistically it is crucial to realise that the level of complexity involved in constructing, using, maintaining, repairing and developing small scale electronic artefacts like the r0ket or especially the door systems is pretty low. Therefore, the necessary expertise and thus the barrier to participation is rather low too. These are not complicated systems in

[151] Discussed in the next section.

engineering terms. Their interesting properties are not derivative of high-tech innovation or any technological breakthrough. Instead, **they are elaborated in a social milieu that encourages experimentation, presents unique problems to participants and expects users to be interested in developing their expertise – however meagre it may be – through understanding artefacts.**

11.1.1 Properties of unfinished artefacts

Facing the extreme variety in various measures of openness within OSHW projects, I introduced the concept of *unfinished artefacts* to refer to small scale electronic artefacts which respond to the hacker ethos. In other words, unfinished artefacts are supposed to be *good* OSHW projects. I noted that the OSHW definition fails to capture the socio-political significance of the r0ket badge. Indeed, licencing the device was literally an afterthought for its designers. To understand the r0ket device adequately it is not enough to focus on the licencing regime and its consequences. On the one hand, it is necessary to consider narrower technical issues of its functional composition. On the other hand, it is necessary to consider the broader social relations that it is part of (see the next Section, 11.1.3).

On the technical issues: there are specific technical choices, functional components and compositional techniques that were involved in the making of the r0ket badge, and that in turn make it a recognisable product of peer production practices. I observed a set of design principles which characterise unfinished artefacts. On the most fundamental level, an unfinished artefact is a simple system that produces complex results. That assertion does not say much about *design principles* though: it is a *design goal* that defines the problem hackers set for themselves. Nonetheless, it already helps to distinguish unfinished artefacts from technological systems that are large from the start and whose performance is deemed critical, such as aeroplanes or nuclear reactors. In the latter cases it is normal to increase complexity for marginal gains in safety, for instance. Of course the same assertion also distinguished unfinished artefacts from trivial systems whose implementation and behaviour is straightforward, such as a doorbell.

In a somewhat haphazard summary of the observations from the two case studies of strategically chosen unfinished artefacts, I conclude that they are *reproducible*, *transparent* and *modular*. It could be argued that the first term concerns their creation, the second everyday use, and the third: subsequent development. I point out three design principles that contribute towards these properties – but without suggesting any sense of comprehensiveness.

Table 5 presents these properties or design principles in a concise manner.

Table 5: Some design principles or properties of unfinished artefacts (a.k.a. good OSHW)

Reproducible	Transparent	Modular
Open licence	Explicit interfaces	Off-the-self parts
Good documentation	Verbose feedback	Loose coupling
Good support	Maintainability	Low-level interfaces

11.1.1.1 Reproducibility entails a number of factors that appear to be external to the artefact. First, I observed the *loose adoption of OSHW and FLOSS licences* to make the reproduction of unfinished artefacts legal. I noted that as far as unfinished artefacts exist in their homeland of the hackerspaces, protecting the rights of users through legal instruments is not a primary concern of practitioners. Second, each project I encountered had a virtual doppelganger, a non-functional representation as a wiki site or wiki page, which suggested that *good documentation* is an important factor. At the same time, I also saw that documentation is by definition out of date with the current implementation. Third, the ultimate source of reproducibility in the hackerspaces context is *good support.*

Some practitioners perceived themselves as working on OSHW simply because they were open to share their experiences and teach others about how to understand, use and build small scale electronic artefacts. In this sense OSHW is not about the material residue of activity but about sharing the expertise and the material infrastructures that are necessary for participating in such peer production practices. Talking to the designers can change licences and yield better documentation. Here it is worth to note that personal exchange with the original creators appeared to be, not only the primary factor in the reproducibility of unfinished artefacts, but also one that seems to be more important for hardware than for software projects, for a number of reasons.

11.1.1.2 Transparency concerns the behaviour of the artefact in its interaction with the user. The very concept of the user is extremely central to the hackers' engineering practices and technological ideals, a fact that can only be appreciated properly in light of the notion promoted by Lialina (2012) that the history of design is the history of the disappearance of the user, who is shifted out of stage. She points out that the discipline which began as Human-Computer Interaction ended up as Experience Design – a process through which both Humans and Computers disappeared in order to merge into an experience of design where the very act of using a technology is lost. Hackers oppose such a trend because they like both Humans and Computers, and cherish the encounter between them which is technology usage.

Therefore, unfinished artefacts are designed to have *explicit interfaces* such as tangible buttons (on both r0ket badges and door systems) rather than automatic sensors that make decisions autonomously. The gesture here is to save human time from performing tedious operations, yet keep humans in control. The radicalisation of such a property is when *all* technical components are exposed – perhaps because any could serve as a potential interface for interacting with the artefact. I noted that the r0ket badge (in conformance to the long tradition of electronic name tags) ships as a "naked" PCB, lacking any cover – which can be interpreted as a critical design gesture targeting mainstream practices of blackboxing commodity electronics. Similarly, the epithet I used for door systems was *dangling wires* to indicate the impression of a temporary construction.

In the same line, unfinished artefacts provide *verbose feedback* to their users, so that a dialogue between humans and non-humans can be initiated. On a door system this is typically manifested as a green and a red LED that correspond to buttons coloured analogously. The lights only turn on when the system has performed the requested operation (e.g. opening or closing the space). The radicalisation of verbose feedback is the common practice of developing chat bots coupled with door systems that participate on the communication platforms of the hackerspace and announce changes in the space state. r0kets broadcast their presence and the messages set by users in a similar way. Both cases involved visualisations of the activity of artefacts that made them more visible to their users.

Finally, transparency involves *maintainability*, e.g. failing in obvious ways that allow repair by non-experts. Even though the lack of a cover on the r0ket may result in damage from environmental factors, the "naked" PCB exposes electronic components so that it is trivial to spot faulty connections, while door systems seldom have plastering over their wires, which helps in finding contact problems. Even in the absence of error messages or other diagnostic feedback, users should be able to look at the artefact and see what is broken. The next design principles make diagnostics and repair more straightforward.

11.1.1.3 Modularity is enhanced by using *off-the-shelf parts* which can be easily obtained on the market, or at least stocked in hackerspaces. Once again, this is a property that is not peculiar to hardware but emphasised in the designs and discourses around OSHW to a much greater degree than in software – no doubt because it is easier to obtain software parts than hardware parts. Using common parts have so many benefits that it is hard to touch on them all. They make it easier to reproduce artefacts following the published design, make it easier to use and develop them if users and developers are familiar with standard components, and make it easier to repair the artefact when something is broken.

Loose coupling, in turn, allows parts to function relatively independently of one another, so that they can be managed separately as much as possible. As before, loose coupling allows users to understand the system part-by-part, and enables them to expand it without breaking things in unexpected places. It is one of the main mechanisms that as Benkler argues enable peer production practices, since once the interfaces between parts are established appropriately, development can happen in parallel and in a distributed fashion. Indeed, no member of the r0ket team knew all about all the ways in which the r0ket have been modified, even if these modifications have been integrated into the default r0ket artefact since. Similarly, hackerspace members working on one end of a door system are sometimes oblivious of work done on the other – for instance between changing the physical interface and adding an extra sensor to the published statistics. Benkler's two other factors (in his terminology *granularity* and *modularity*) that enable a wide range of small and big as well as the easy integration of results to the commonly developed product are also greatly enhanced by loose coupling.

Last but not least, exposing *low-level interfaces* to users is perhaps the most peculiar property of unfinished artefacts.

11.1.2 Theoretical consequences

Such a set of design principles – reproduced in variety forms by a various researchers of hacker culture – gives an adequate account of peer production practices as well as the artefacts they work with. However, it does not draw the theoretical consequences of these practices in a form that is useful for the general understanding of technology in terms of Science and Technology Studies. Therefore, I proposed a critique of the SCOT model that extends it in a way that it can accommodate unfinished artefacts. I hope that such a critique can stand as a contribution to the understanding of technology in terms of Science and Technology Studies in general.

My argument is that unfinished artefacts include functional components in their technical composition which fend off stabilisation and closure, preserve interpretative flexibility, and contribute positively to the reliability of the given technology. I called these elements opening mechanisms, conjectured that they can be retrofitted to technological artefacts with enough work, characterise unfinished artefacts at their best, and can be taken away in order to finish an unfinished artefact: unfinished artefacts can be finished, but do not tend towards being finished. Indeed, the observation in relation to both small scale electronic hardware – the r0ket badge and the door systems – was that unfinished artefacts that see continuous use tend to expand into large technological systems in the sense of Hughes (2012). The argument about opening mechanisms in unfinished artefacts was largely based on the fact that they expose low-level interfaces which can intervene in the functionality of the device. However, I suggest that a similar argument could be made – perhaps in a less straightforward manner – using any or all of the design principles listed above. **The results obtained from unfinished artefacts suggest that the classic SCOT assumption about decreasing interpretative flexibility through closure mechanisms corresponds to increasing reliability, functionality and consensus can be revised.**

On another note, – as expected from the previous literature addressing the hypothesis about the generality of peer production – the design principles as well as their theoretical consequences seem to be applicable across software and hardware with appropriate shifts in emphases. I argued that the greater role of fixed capital in hardware hacking orients developers towards vendors who are not necessarily participating in the peer production process, whereas in the case of software, practitioners are more dependent on each other. Hardware vendors cannot choose to participate in peer production practices as "just another developer" in the way that IBM programmers work on the Linux kernel for instance, because the former play a different role in the work flow that can be hardly replaced by simple members of the developer community. Therefore, the relationship between the community and the industry is more complicated in the peer production of hardware than in the peer production of software.

On the other hand, the necessity of separate tangible reproductions of the design encourages independent implementations and therefore divergences in design so that family resemblances emerge more easily than incremental developments of a single design. Since telematics plays less part in the process, designs often adapt to the local context, from the availability of spare parts to the particular needs of users, and improvements do not necessarily make it back "upstream" to the original contributor of the project. In Benkler's analytical framework it could be argued that it is harder to integrate contributions to hardware designs than to the source code of software.

As the above exposé clearly shows, there are many tenets of critical design practices that hackers follow, even if their meaning and interpretation is obscure at best, and contradictory at worst. There are three ways in which hackers deal with the ambiguity of design principles. One is that principles are complemented by aphorisms, patterns and "best practices" that serve as the middle range theory of hackers and passed down through generations as part of the hacker lore. The other is improvisation. Faced with theoretical difficulties, many hackers are ready to give up their principles and use readily available materials or techniques to solve the problem at hand.

Finally, these two contradictory mitigation strategies meet in an elusive sense of engineering aesthetics – one which good hackers are expected not only to be able to execute in the material but also to elaborate lucidly to others as discourse. Moreover, the engineering aesthetics of hackers overlaps at several points with mainstream engineering standards, but shows significant differences too. In order to reproduce these differences from one generation to the next and to enforce them amongst the participants of the scene, hackers are engaged in an endless conversation and critique that revolves around technological artefacts of the most varied kinds. Hackerspaces contain artefacts that are thought to display prominent examples of these values and guard the hacker tradition of engineering in a conservative fashion. Museums of working computers are reminiscent of such a trend.[152]

[152] Notable connections exist at a diverse number of location from Amersfoort, The Netherlands (at Hack42) through Palazzolo Acreide, Sicily (associated with Freaknet hacklab) to in Montréal, Canada (at Foulab).

11.1.3 Unfinished architectures as the homeland of unfinished artefacts

Neither the instrumental use nor the social scientific understanding of unfinished artefacts is possible without taking into consideration the social relations which produced them and in the context of which they are functional and meaningful. Mainstream technology like commodity electronics is designed to work under a wide range of social conditions and make sense in the context of various infrastructures – or at least those social conditions are widespread enough and the infrastructures standardised enough that commodity electronics can function. With unfinished artefacts it is not necessarily so: while a mobile phone is useful because a lot of people has a mobile phone, the r0ket badge as a communication device is most useful in the midst of a hacker convention when almost all participants wield one. Similarly, everybody understands the concept of opening time, but a door system that provides a real time indication of space state is only useful for people who can spontaneously change plans over the course of their day, which entails certain privileges. More importantly, the emphasis in the design of unfinished artefacts is not on reliability achieved through blackboxing (so that users cannot tamper with its internals) but on maintainability (so that it can be fixed when it breaks). However, maintainability requires a baseline of expertise from all participants. Even though the other design principles like simplicity, transparency and responsiveness lower the barrier of entry because they make it easier for users to understand what the device is doing or what it is trying to do, it often still requires some expertise to know where to look for clues. The exact social conditions that unfinished artefacts rely on, of course, are summarised under the rubric of unfinished architectures.

11.2 From peer production practices to unfinished architectures

Asking what are the social relations that unfinished artefacts produce and what are the social conditions under which it is possible to produce unfinished artefacts lead to an extensive research into the prehistory (hacklabs), history and current social dimensions of hackerspaces. I referred the results of these investigations under the category of *unfinished architectures*, to denote the fairly stable tradition of hacker clubs and the ways in which they allow for a continuous yet heterogeneous expression of undirected technological creativity through unalienated labour. **Unfinished architectures expand the definition of peer production practices in order to theorise social norms and customs of collaboration together with material infrastructures and historically/geographical situated contexts which enable the production of commons.**

I presented a panorama of social functions performed by the two unfinished artefacts under consideration – the r0ket badge and the door systems. I showed that enumerating the technical components of the r0ket badge amounts to a tour of the various streams of hackers cultures that come together on the site of hackerspaces and hacker conventions. Just like the hacker conventions are a material condensation of the hackers' world, the r0ket badge encapsulates the whole microcosm of the hackerspaces as a material trace of human activities. Conversely, the r0ket badge itself "calls forth a people" (in the wording of Heidegger) – that is, contributes to the reproduction of the scene through gathering all its properties into a significant form. Similarly, door systems play a symbolic role in delineating the perimeters of hackerspaces. Their marked social role is evident in the theatrical technological rhetorics of the spatial presentation of their tangible interfaces. Individuals become hackers through contributing to door systems, and new members earn the right to toggle the space state that allows members of the public to know when they can enter the laboratory. Door systems play an analogous role in connecting new hackerspaces to the network of existing laboratories. By implementing a door system the new hackerspace becomes similar to other hackerspaces and therefore recognisable as one of them. Furthermore, by exposing the space state through a standard interface common to a lot of hackerspaces, the new hackerspace can integrate into the large technological infrastructure that is the backbone of internal communication flows between hackerspaces. The latter is not restricted to the flow of information as such, but manifests as an increase in the number of visitors into the tangible space as well.

These functions – symbolic and technical at once – are all the more important because there are no formal requirements or bureaucratic formalities for becoming a hackerspace. Therefore, beyond the self-nomination of registering on the hackerspaces.org website which aggregates contact information, hackerspaces have to earn the label through their practical achievements. Once again, the process closely follows the trajectory of hackers who are not reliant on official certifications for their identity but who have to earn their name in the scene through their contributions.

The case studies of the r0ket badge and the door systems showed how hackers *work through materials* on two objects simultaneously. On the one hand, they construct small scale electronic artefacts that are interesting in themselves as far as the values particular to their engineering culture into them. This is why I proposed to call those *unfinished artefacts*. On the other hand, they intervene in the social relations that constitute the internal relations of the hackerspaces scene as well as the relations between the community and the industry, reproducing what I called their relative autonomy. I proposed to call these environments *unfinished architectures*.

By relating peer production practices as they are performed in the hackerspaces to their historical context, I tried to dispel some widespread misrepresentations and misunderstandings of these phenomena. Too many studies set out from the premise that hacking is a form of collective action exploiting the unintended consequences of the development in the productive forces, e.g. Information and Communication Technologies. In such literature hackers sometimes appear to make an untimely intervention in history, entering a totality from the outside. In the other extreme, they are able to subvert structural constraints because they have single-handedly created the technical architectures and large technological systems that form the basis of social relations today. In my opinion both ways to approach the problem lend too much agency to hackers.

In order to counterweight these one-sided interpretations, I argued that hackerspaces exhibit a remarkable continuity with their historical period, be it meant to be an epoch such as modernity, capitalism and liberal democracy in general, or a more specific historical era such as late modernity, cognitive capitalism and neoliberalism. Their conditions of emergence as well as the actual process of their social formation have been the result of decades of development, and there is no reason to assume that the future trajectory will break with structural constraints. As much as hackers can be seen as disruptive of these structural logics can be seen in through the social construction of their relative autonomy as a specific social group. **Finally, through the *technical interrogation* and *object biography* of small scale electronic artefacts that can be considered the material residue of these small and big histories, I have shown how social conflicts and contradictions leave their mark on the actual technical output of hackerspaces.**

12 Afterword

The results open up several perspectives for further research. Three may be worthy of consideration. These are briefly described in the next paragraphs.

First, it would be possible to **deepen the analysis through considering the engineering subculture of hackers and the sites of hackerspaces in light of the relationship between practice and theory that they articulate**, along the lines of Pickering and Guzik (2008) and Pickering (2010). Such research would give an opportunity to ground the properties of unfinished artefacts outlined in the previous section in a coherent framework. The hackerspaces are interesting as a non-institutional setting for the cultivation of engineering expertise, where the ontological conceptions of engineering work can be closer reflected in the organisational and management of the laboratory. Phenomena identified by Pickering like the mangle, strange performances, and ontological theatres seem to manifest in unfinished artefacts and architectures. This line of investigation could open possibilities for the reconsideration of computer science history as well, seeing how hackers sustain and pick up thwart trajectories such as biological computers, hypnosis or forgotten system architectures.

Second, it would be possible to **extend the analysis through following hackers in the hackerspaces scene to the field of the life sciences**. As already mentioned, DIY biology as the latest addition to the technological repertoire of hackers reconfigures almost all variables discussed in the dissertation. The relationship between the community and the industry is certainly reconfigured with the arrival of biology, which tried hackers' abilities to rearticulate the relative autonomy of their social group in a different socio-technical environment. Biotechnologies obviously build on advances in both hardware and software, so that the relationship of these three kinds of technologies could be reconsidered in terms of the peer production practices of practitioners. It is interesting to see how the alternative conception of technology that hackers cultivated for decades meets one of the most conservative hard science disciplines. Delfanti's pioneering work on biohacking could serve as a starting point for such an investigation (2013).x

Third, there are **ample opportunities for comparison with other social groups which cultivate an idiosyncratic relationship to technology**. The basis of such work could be the affection hackers express for old technologies in general, and retro-computing in particular. This little investigated area of hacker culture has many points of affinity with Luddites old and new. Classic machine breakers who say the advent of industrial capitalism were craftsmen in defence of the form of life allowed by the cottage industry, echoing sentiments of disillusioned freelance web designers who have to face the rise of media monopolies in the World Wide Web, including the professionalisation of digital labour. Some connections have already been traced between the Luddites and the hacker scene by Sale (1996). Neo-primitivist movements today are inspired by the critique of the Unabomber that the "Industrial Revolution and its consequences have been a disaster for the human race" (Kaczynski 1995), echoing many common complaints heard in the hackerspaces about the large technological systems managed by modern institutions. Could the gestures of these social groups in face of a runaway technology related to the puritan aesthetics of text only command line interfaces that hackers stick to since the birth of the personal computer? Hackers' resistance to the technological imperative (Ellul 1980), such as the widespread resistance to mobile phones in the hackerspaces milieu, could be put in a wider historical context. Hackers' often radical choices in terms of the development, adoption and use of technologies could be considered vis-a-vis their gate keeping and community building practices, which could relate them to critical adoption and non-adoption within the Amish.

Such investigations could further clarify how hackers rearticulate – and sometimes subvert – the prevailing contradictions of liberalism, capitalism and modernity.

References

Ackermann, John R. 2009. “Toward Open Source Hardware.” *University of Dayton Law Review* 34 (2) (Winter): 183–223. https://www.tapr.org/Ackermann_Open_Source_Hardware_Article_2009.pdf.

Adams, Douglas. 1985. *The Original Hitchhiker Radio Scripts.* Ed. Geoffrey Perkins. London: Pan Books.

Administration, Federal Aviation. 2015. “DOT and FAA Propose New Rules for Small Unmanned Aircraft Systems: Regulations Will Facilitate Integration of Small UAS into U.S. Aviation System.” Press release. http://www.faa.gov/news/press_releases/news_story.cfm?newsId=18295.

Adorno, Theodor. 1976. “Introduction.” In *The Positivist Debate in German Sociology*, ed. Theodor Adorno. London: Heinemann.

Agrawal, Dr. Nitin, Dr., YassinA. Hassan, and VictorM. Ugaz. 2007. “A Pocket-Sized Convective PCR Thermocycler.” *Angewandte Chemie International Edition* 46 (23) (June): 4316–4319. doi:10.1002/anie.200700306. http://onlinelibrary.wiley.com/doi/10.1002/anie.200700306/abstract.

Aibar, Eduard, and Maxigas. 2014a. “Closure and Stabilization in Open Source Artefacts.” Presentation at the SCOT2014 doctoral seminar, Trondheim, Norwegian University of Science and Technology. http://www.ntnu.edu/kult/scot2014.

———. 2014b. “From Open Science to Open Technology / de La Ciencia Abierta a La Tecnología Abierta.” *Argumentos de Razón Técnica* (17) (June): 115–136. institucional.us.es/revistas/argumentos/17/art_7.pdf.

Akrich, Madeleine. 1992. “The de-Scription of Technical Objects.” In *Shaping Technology / Building Society: Studies in Sociotechnical Change*, ed. Wiebe Eco Bijker and John Law, 205–224. Cambridge, MA: MIT Press.

Akrich, Madeleine, and Bruno Latour. 1992. “A Summary of a Convenient Vocabulary for the Semiotics of Human and Nonhuman Assemblies.” In *Shaping Technology / Building Society: Studies in Sociotechnical Change*, ed. Wiebe Eco Bijker and John Law. Cambridge, MA: MIT Press.

Altman, Mitch. 2012a. “The Hackerspaces Movement.” Ted talk. http://www.tedxbrussels.eu/2012/speakers/mitch_altman.php.

———. 2012b. “Hacking at the Crossroad: US Military Funding of Hackerspaces.” *Journal of Peer Production* (2). http://peerproduction.net/issues/issue-2/invited-comments/hacking-at-the-crossroad/.

ana. 2004. “Hacklabs: From Digital to Analog.” Blog entry, translated from the Suburbia Telemacktical MediaZine. https://network23.org/ana/hacklabs-from-digital-to-analog/.

Anarchopedia contributors. 2006. “LOA Hacklab.” Encyclopedia entry.

Anderson, Chris. 2006a. *The Long Tail: Why the Future of Business Is Selling Less of More*. New York: Hyperion. http://libgen.in/book/index.php?md5=D419ECDE90E42A5A53CF63E5F9425C5C.

———. 2006b. "People Power: Blogs, User Reviews, Photo-Sharing – the Peer Production Era Has Arrived." Article in Wired Magazine 14.07. http://archive.wired.com/wired/archive/14.07/people.html.

———. 2009. *Free: The Future of a Radical Price*. New York: Hyperion. http://libgen.in/book/index.php?md5=6EAB515A5F3088D5CFACA9177ECA196C.

———. 2014. *Makers: The New Industrial Revolution*. New York: Crown Business. http://libgen.in/book/index.php?md5=fc72dd418924b97acfc2cfe3da7e16b6.

anonymous. 2010. "Storia." Wiki page. http://web.archive.org/web/20100613015928/http://hackmeeting.org/hackit09/index.php?page=storia&lang=en.

———. 2011. "Geigercounter." Page on the wiki of the 28th Chaos Communication Camp (28C3). http://events.ccc.de/congress/2011/wiki/Geigercounter#Connecting_the_Geiger_Counter_to_your_r0ket.

Antonić, Vojislav "Voja". 2014. "Vojislav Antonić in Calafou." Talk in Calafou, Catalunya, Spain. https://calafou.org/en/content/voja-antonic-calafou.

Appelbaum, Jacob. 2011. "The Coming War on General Computation." Talk at 29C3, The 29th Chaos Communication Congress. http://events.ccc.de/congress/2011/Fahrplan/events/4848.en.html.

Arvidsson, Adam, and Nicolai Peitersen. 2013. *The Ethical Economy: Rebuilding Value After the Crisis*. New York: Columbia University Press.

Association, Open Source Hardware. 2008. "Open Source Hardware (OSHW) Definition 1.1." http://freedomdefined.org/Definition.

Aust, Stefan. 2008. *Baader-Meinhof: The Inside Story of the R.A.F.* London: Bodley Head.

Autistici/Inventati. 2012. *+kaos: 10 Anni Di Hacking E Mediattivismo*. Ed. Laura Beritelli. Milan: Agenzia X. http://www.autistici.org/it/who/book.html.

Avana.net contributors. 2005. "Progetto Ultra Lab Al Forte." Web page. http://avana.forteprenestino.net/ultralab.htm.

Babbie, Earl. 2010. *The Practice of Social Research*. Belmont, CA: Wadsworth, Cengage Learning.

Baker, Ulus. "A Comment on Dziga Vertov: The Cine-Eye." *Körotonomedya*. http://www.korotonomedya.net/kor/index.php?id=21,181,0,0,1,0.

Balázs, Béla. 1998. *Halálesztétika [Death Aesthetics]*. Budapest: Magvető.

Balka, Kerstin, Christina Raasch, and Cornelius Herstatt. 2009a. "Open Source Enters the World of Atoms: A Statistical Analysis of Open Design." *First*

Monday 14 (11) (November). http://firstmonday.org/ojs/index.php/fm/article/view/2670/2366.

———. 2009b. "Open Source Beyond Software: An Empirical Investigation of the Open Design Phenomenon." Paper presented at the R&D Management Conference, Feldafing near Munich, Germany.

———. 2010. "Open Source Innovation: A Study of Openness and Community Expectations." Paper presented at the DIME Conference, Milan, Italy.

Ball, Matt. 2012. "TechShop Announces Partnership and Co-Location with Lowe's in Austin Area." News article. http://informedinfrastructure.com/2145/techshop-announces-partnership-and-co-location-with-lowes-in-austin-area/.

Banzi, Massimo. 2008. *Getting Started with Arduino.* 1st ed. O'Reilly Media / Make:Books.

Barbrook, Richard. 2007. *Imaginary Futures: From Thinking Machines to the Global Village.* London: Pluto Press. http://libgen.in/book/index.php?md5=E5849670CB06BF01E125F9E52ABA22F8.

Barbrook, Richard, and Andy Cameron. 1996. "The Californian Ideology." *Science as Culture* 26: 44–72. http://www.imaginaryfutures.net/2007/04/17/the-californian-ideology-2.

Barlow, John Perry. 1996a. "Crime & Puzzlement." In *High Noon on the Electronic Frontier: Conceptual Issues in Cyberspace*, ed. Peter Ludlow, 459–486. London: MIT Press.

———. 1996b. "A Declaration of the Independence of Cyberspace." https://projects.eff.org/~barlow/Declaration-Final.html.

Barron, Anne. 2013. "Free Software Production as Critical Social Practice." *Economy and Society* 42 (4): 597–625. doi:10.1080/03085147.2013.791510. http://libgen.in/scimag/index.php?s=10.1080/03085147.2013.791510.

Bataille, George. 1991. *The Accursed Share.* Vol. 1. New York: Zone Books.

Bauman, Zygmunt. 1991. *Modernity and Ambivalence.* Cambridge: Polity Press. http://libgen.in/book/index.php?md5=24ED90CDC03245AE203D092CD48C46B4.

Bauwens, Michel. 2005. "The Political Economy of Peer Production." *CTHEORY* (December). http://www.ctheory.net/articles.aspx?id=499.

———. 2012. "Blueprint for P2P Society: The Partner State & Ethical Economy." *Shareable: Work and Enterprise* (July). http://www.shareable.net/blog/a-blueprint-for-p2p-institutions-the-partner-state-and-the-ethical-economy-0.

Bazichelli, Tatiana. 2008. *Networking: The Net as Artwork.* Ed. Simonetta Fadda. Aarhus: Aarhus Universität.

Beaulieu, Anne, Andrea Scharnhorst, and Paul Wouters. 2007. "Not Another Cast Study: A Middle-Range Interrogation of Ethnographic Case Studies in the Exploration of E-Science." *Science, Technology & Human Values* 32 (6) (November): 672–692.

Becha. 2012. "Hackerspaces Exchange." https://events.ccc.de/congress/2012/wiki/Hackerspaces_exchange.

Beck, Ulrich. 1992. *Risk Society: Towards a New Modernity.* New Delhi: Sage.

Beck, Ulrich, Wolfgang Bons, and Christoph Lau. 2003. "The Theory of Reflexive Modernization." *Theory, Society and Culture* 20 (2): 1–33. doi:10.1177/0263276403020002001. http://tcs.sagepub.com/content/20/1/133.short.

Beck, Ulrich, Anthony Giddens, and Scott Lash. 1994. *Reflexive Modernization: Politics, Tradition and Aesthetics in the Modern Social Order.* Palo Alto, CA: Standord University Press.

Beckers, Dennis, and Peter van den Besselaar. 2014. "The Tales of the Three Digital Cities of Amsterdam: The Application of ICT for Social and Political Participation." *Communication Management Quarterly* (30): 105–130. doi:10.5937/comman1430115B.

Beito, David T. 2000. *From Mutual Aid to the Welfare State: Fraternal Societies and Social Services, 1890-1967.* Cambridge, MA: University of North California Press. http://libgen.in/book/index.php?md5=6F132D55788F475CEB1351D6B2ED2974.

Bell, Daniel. 1973. *The Coming of the Postindusrial Society: A Venture in Social Forecasting.* New York: Basic Books.

Bell, Genevieve, and Paul Dourish. 2007. "Back to the Shed: Gendered Visions of Technology and Domesticity." *Personal and Ubiquitous Computing* 11 (5) (June): 373–381. http://www.dourish.com/classes/readings/BellDourish-Sheds.pdf.

Benchoff, Brian. 2013. "3D Printering: Key Patents." Blog entry. http://hackaday.com/2013/09/11/3d-printering-key-patents/.

———. 2015. "LayerOne Demoscene Board." Blog entry. http://hackaday.com/tag/demoscene/.

Benkler, Yochai. 2002. "Coase's Penguin, or, Linux and 'the Nature of the Firm'." *The Yale Law Journal* 112 (3) (December): 369–446. doi:10.2307/1562247. http://libgen.in/scimag/?s=10.2307/1562247.

———. 2006. *The Wealth of Networks: How Social Production Transforms Markets and Freedom.* New Haven, CT: Yale University Press.

Benkler, Yochai, and Helen Nissenbaum. 2006. "Commons-Based Peer Production and Virtue." *The Journal of Political Philosophy* 14 (4): 394–419. http://www.nyu.edu/projects/nissenbaum/papers/jopp_235.pdf.

Bernard, Harvey Russel. 2006. *Research Methods in Anthropology: Qualitative and Quantitative Approaches.* 4th ed. Lanham, MD; New York; Toronto; Oxford: AltaMira Press.

Besselaar, Peter van den, and Satoshi Koizumi. 2003. “The Life and Death of the Great Amsterdam Digital City.” In *Digital Cities III. Information Technologies for Social Capital: Cross-Cultural Perspectives – Third International Digital Cities Workshop, Amsterdam, the Netherlands, September 18-19, 2003. Revised Selected Papers*, ed. P. van den Besselar and S. Koziumi, 3081:66–96. Lecture Notes in Computer Science. Berlin; Heidelberg: Springer-Verglag. doi:10.1007/11407546_4. http://link.springer.com/chapter/10.1007%2F11407546_4.

Bey, Hakim. 1991. *T.A.Z.: The Temporary Autonomous Zone, Ontological Anarchy, Poetic Terrorism.* New York: Autonomedia.

Bijker, Wiebe E. 1995. *Of Bicycles, Bakelites, and Bulbs: Toward a Theory of Sociotechnical Change.* Cambridge, MA: MIT Press.

Bijker, Wiebe Eco, and John Law, ed. 1992. *Shaping Technology / Building Society: Studies in Sociotechnical Change.* Cambridge, MA: MIT Press.

Blanc, Sabine, and Ophelia Noor. 2011. “30 Years of Political Hacking.” Augmented news article on Owni. http://owni.eu/2011/11/08/30-years-of-political-hacking/.

Bobe, Jason. 2008. “Don’t Phage Me, Bro.” Blog entry on DIYbio. http://diybio.org/2008/05/24/dont-phage-me-bro/.

Boltanski, Luc, and Ève Chiapello. 2005. *The New Spirit of Capitalism.* London; New York: Verso.

Boltanski, Luc, and Laurent Thévenot. 2006. *On Justification: The Economies of Worth.* Princeton, NJ: Princeton University Press.

Borburn, Jason. 2005. *Street Science: Community Knowledge and Environmental Health Justice.* Cambridge, MA: MIT Press.

Borland, John. 2007. “‘Hacker Space’ Movement Sought for U.S.” News article, WIRED magazine. http://www.wired.com/2007/08/us-hackers-moun/.

Boudon, Raymond. 1991. “What Middle-Range Theories Are.” *Contemporary Sociology* 20 (4): 519–522.

Bourdieu, Pierre. 1977. *Outline of a Theory of Practice.* Cambridge; New York: Cambridge University Press.

Boutang, Yann Moulier. 2011. *Cognitive Capitalism.* Cambridge: Polity Press.

Boyle, James. 2009. “Bibliography on Peer Production.” Wiki page. http://p2pfoundation.net/Bibliography_on_Peer_Production.

———. 2013. “The Second Enclosure Movement and the Construction of the Public Domain.” Article on personal website. http://www.law.duke.edu/journals/66LCPBoyle.

Brainsmoke. 2014. “PCB Techinc Logo.” Wiki page. https://wiki.techinc.nl/index.php/PCB_Techinc_Logo.

Braybrooke, Kat. 2011. “She-Hackers: Millennials and Gender in European F/LOSS Subcultures — a Presentation of Research and Invitation for Debate.” Talk at the 4th Chaos Communication Camp, organised by the Chaos Computer Club, Finowfurt airport. http://events.ccc.de/camp/2011/Fahrplan/events/4487.en.html.

Bre, and Astera, ed. 2008. “Hackerspaces: The Beginning.” ebook. http://blog.hackerspaces.org/2011/08/31/hackerspaces-the-beginning-the-book/.

Brody, Richard. 2008. *Everything Is Cinema: The Working Life of Jean-Luc Godard*. New York: Metropolitan Books.

Brooks, Frederick. 1975. *The Mythical Man-Month: Essays on Software Engineering*. Addison-Wesley.

Buchanan, R. A. 1983. “Gentleman Engineers: The Making of a Profession.” Presented at a seminar in the School of Social Sciences at the Australian National University, Canberra, in April 1981. http://libgen.in/scimag/get.php?doi=10.0000%2Fjstor.org%2F21105520214911.

Bylund, Per. 2014. “Was Ronald Coase an Austrian?” Blog entry for the Mises Institute. http://bastiat.mises.org/2014/07/was-ronald-coase-an-austrian/.

Caldas, Alexandre, Paul A. David, and Orges Ormanidhi. 2005. “Digital Information Network Technologies, Organisational Performance and Productivity: An Exploratory Study of the Public Sector in Europe.” Research project report. http://www.oii.ox.ac.uk/resources/publications/OIIRP1_200512_Report.pdf.

Callon, Michel. 1986. “The Sociology of an Actor-Network: The Case of the Electric Vehicle.” In *Mapping the Dynamics of Science and Technology: Sociology of Science in the Real World*, ed. Michel Callon, John Law, and Arie Rip, 29–30. London: Macmillan.

Carlsson, Anders. 2009. “The Forgotten Pioneers of Creative Hacking and Social Networking: Introducing the Demoscene.” In *Re:live: Media Art Histories (Refereed Conference Papers)*, ed. Sean Cubitt and Paul Thomas, 16–20. Melbourne: University of Melbourne; Victorian College of the Arts; Music.

Casalegno, Federico. 1999. “Comunitarian Dynamics, Electro-Electives Affinities and Networked Memories in the Contemporary Cyberculture: The Nettime List. Interview with Patrice Riemens.” Interview for the Living Memory project. http://web.media.mit.edu/~federico/living-memory/english/interview_riemens.html.

Castells, Manuel. 1996. *The Rise of the Network Society, the Information Age: Economy, Society and Culture*. Vol. I. Cambridge, MA; Oxford: Blackwell.

———. 1997. *The Power of Identity, the Information Age: Economy, Society and Culture*. Vol. II. Cambridge, MA; Oxford: Blackwell.

———. 1998. *End of Millennium, the Information Age: Economy, Society and Culture*. Vol. III. Cambridge, MA; Oxford: Blackwell.

———. 2009. *Communication Power*. Oxford: Oxford University Press.

———. 2012. *Networks of Outrage and Hope*. Cambridge: Polity Press.

———. 2014. "Interdisciplinary Analysis of the Network Society." Online course at Universitat Oberta de Catalunya.

Cavalcanti, Gui. 2014. "Is It a Hackerspace, Makerspace, TechShop, or FabLab?" http://makezine.com/2013/05/22/the-difference-between-hackerspaces-makerspaces-techshops-and-fablabs/.

CERN. 2012. "Open Hardware Repository Manifesto." http://www.ohwr.org/projects/ohr-support/wiki/Manifesto.

Chan, Anita Say. 2014. "Beyond Technological Fundamentalism: Peruvian Hack Labs & 'Inter-Technological' Education." *Journal of Peer Production* (5) (June). http://peerproduction.net/issues/issue-5-shared-machine-shops/peer-reviewed-articles/beyond-technological-fundamentalism-peruvian-hack-labs-and-inter-technological-educat

Cherney, Max. 2014. "The FBI Says It Can't Find Hackers to Hire Because They All Smoke Pot." Motherboard Vice Channel article. http://motherboard.vice.com/read/the-fbi-cant-find-hackers-that-dont-smoke-pot.

Clark, David D. 1992. "A Cloudy Crystal Ball - Visions of the Future." In *Proceedings of the Twenty-Fourth Internet Engineering Task Force*, ed. Megan Davies, Cynthia Clark, and Debra Legare, 539–544. Cambridge, MA: Internet Engineering Task Force; MIT NEARnet.

Clark, Melody, and Ricardo Gomez. 2012. "Libraries, Telecenters Cybercafés: A Comparison of Different Types of Public Access Venues." In *Libraries, Telecenters, Cyberafes and Public Access to ICT: Interntaional Comparisons*, ed. Ricardo Gomez, 1–10. Hershey, PA: IGI Global.

Cleaver, Harry. 1992. "The Inversion of Class Perspective in Marxian Theory: From Valorization to Self-Valorization." In *Essays on Open Marxism*, ed. Werner Bonefeld, Richard Gunn, and Kosmas Psychopedis. London: Pluto Press.

Clement, Andrew, and Leslie Regan Shade. 1999. "The Access Rainbow: Conceptualizing Universal Access to the Information/Communications Infrastructure." In *Community Informatics: Enabling Communities with Information and Communications Technologies*, ed. Michael Gurstein, 32–51. Hershey, PA; London: IGI. http://www.gbv.de/dms/goettingen/312306776.pdf.

CNN. 2012. "The Revealer: 3-d Violin." News item. http://backstory.blogs.cnn.com/category/the-revealer/.

Coase, Ronald H. 1937. "The Nature of the Firm." *Economica* (4): 385–405.

Coleman, Gabriella. 2010. "The Hacker Conference: A Ritual Condensation and Celebration of a Lifeworld." *Anthropological Quarterly* 83 (1) (Winter): 47–72. http://muse.jhu.edu/journals/anq/summary/v083/83.1.coleman.html.

———. 2012. *Coding Freedom: The Ethics and Aesthetics of Hacking*. Princeton, NJ: Princeton University Press.

———. 2014. *Hacker, Hoaxer, Whistleblower, Spy: The Many Faces of Anonymous*. London; New York: Verso. http://monoskop.org/File:Coleman_Gabriela_Hacker_Hoaxer_Whistleblower_Spy_The_Story_of_Anonymous.epub.

Coleman, Gabriella, and Alex Golub. 2008. “Hacker Practice: Moral Genres and the Cultural Articulation of Liberalism.” *Anthropological Theory* 8 (3) (September): 255–277.

Colle, Royal D. 2004. “ICTs, Telecenters and Community Development.” WSIS research paper. http://wsispapers.choike.org/ict_telecenters_dev.pdf.

Colle, Royal D. 1999. “Communication Shops and Telecenters in Developing Nations.” In *Community Informatics: Enabling Communities with Information and Communications Technologies: Enabling Communities with Information and Communications Technologies*, ed. Michael Gurstein, 415–445. Hershey, PA; London: IGI. http://www.gbv.de/dms/goettingen/312306776.pdf.

Collins, Harry, and Robert Evans. 2007. *Rethinking Expertise*. Chicago; London: University of Chicago Press.

Crockford, Douglas. 2006. “The Application/json Media Type for JavaScript Object Notation (JSON).” *Request for Comments* (4627) (July). https://tools.ietf.org/html/rfc4627.

Cruickshank, Leon. 2014. *Open Design and Innovation: Facilitating Creativity in Everyone*. Farnham: Gower.

CrunchBase. 2014. “3D Robotics.” CrunchBase company database profile. http://web.archive.org/web/20140822003650/http://www.crunchbase.com/organization/3d-robotics.

Culprit, Carrier. 1987. “A Guide to the PRIMOS Operating System.” Article in The Lod/H Technical Journal. http://www.textfiles.com/magazines/LOD/lod-2.

Cuninghame, Patrick. 2005. “Autonomia in the 1970s: The Refusal of Work, the Party and Power.” *Cultural Studies Review* 11 (2) (September): 77–94. doi:http://dx.doi.org/10.5130/csr.v11i2.3660. http://epress.lib.uts.edu.au/journals/index.php/csrj/article/view/3660.

Curtis, Adam. 2011. “All Watched over by Machines of Loving Grace.” Documentary series, BBC. https://thepiratebay.se/torrent/6457676/BBC_All_Watched_Over_By_Machines_Of_Loving_Grace.

Dadusc, Deanna, and ETC Dee. 2015. “The Criminalisation of Squatting: Discourses, Moral Panics and Resistances in the Netherlands and England and Wales.” In *Moral Rhetoric and the Criminalisation of Squatting: Vulnerable Demons?*, ed. Lorna Fox, O’Mahony, David O’Mahony, and Robin Hickey, 109–132. London: Routledge.

Dafermos, George. 2014. “Distributed Manufacturing: Commons-Oriented Productive Capacities.” Policy paper by the FLOK Society. http://floksociety.org/docs/Ingles/2/2.4.pdf.

DARPA, Tactical Technology Office. 2010. "Manufacturing Experimentation and Outreach." Broad Agency Announcement. https://www.fbo.gov/spg/ODA/DARPA/CMO/DARPA-BAA-11-19/listing.html.

DeLanda, Manuel. 2001. "Open-Source: A Movement in Search of a Philosophy." Presented at the Institute for Advanced Study, Princeton, NJ. http://www.cddc.vt.edu/host/delanda/pages/opensource.htm.

Deldanti, Alessandro. 2014. "Is Do-It-Yourself Biology Being Co-Opted by Institutions?" In *Meta Life. Biotechnologies, Synthetic Biology, Alife and the Arts*, ed. Annick Bureaud, Roger F. Malina, and Louise Whiteley. Kindle. Leonardo Ebook Series. Cambridge, MA: MIT Press.

Deleuze, Gilles. 1992. "Postscript on the Societies of Control." *October* 59 (Winter): 3–7. http://links.jstor.org/sici?sici=0162-2870%28199224%2959%3C3%3APOTSOC%3E2.0.CO%3B2-T.

Delfanti, Alessandro. 2013. *Biohackers: The Politics of Open Science*. London: Pluto Press.

Delst, Malcolm van. 2012. "Chaos Computer Club and the Rise of Hacker Culture." Blog entry. http://www.civicactions.com/blog/2012/jan/02/chaos_computer_club_and_the_rise_of_hacker_culture.

Denning, Dorothy E. 1996. "Concerning Hackers Who Break into Computer Systems; Postscript, June 11, 1995." In *High Noon on the Electronic Frontier: Conceptual Issues in Cyberspace*, ed. John Perry Barlow, 137–164. Cambridge, MA: Routledge.

Dickel, Sascha, Jan-Peter Ferdinand, and Ulrich Petschow. 2014. "Shared Machine Shops as Real-World Laboratories." *Journal of Peer Production* (5) (July).

Dicks, Bella, Bruce Mason, Amanda Coffey, and Paul Atkinson. 2005. *Qualitative Research and Hypermedia: Ethnography for the Digital Age*. London; Thousand Oaks, CA; New Delhi: SAGE.

Directa.cat. 2014. "El Juez Gómez Bermúdez Envia a La Presó Set de Les Onze Persones Detingudes Durant L'operació Pandora: L'ús Del Servidor Riseup.net O La Lectura Del Llibre 'Contra La Democràcia' Són Dos Dels Indicis Emprats a La Interlocutòria Per Justificar El Seu Empresonament." News article. https://directa.cat/jutge-gomez-bermudez-envia-preso-set-de-onze-persones-detingudes-durant-loperacio-pandora.

Doctorow, Cory. 2012. "The Coming Civil War over General Purpose Computing." *Boing Boing* (August). http://boingboing.net/2012/08/23/civilwar.html.

Dominick, Brian A. 2014. "An Introduction to Dual Power Strategy." http://left-liberty.net/?p=265.

Dominik, Carsten, and others. 2010. *The Org Mode 7 Reference Manual (for Org Version 7.3)*. Network Theory. http://www.network-theory.co.uk/org/manual/.

Dougherty, Dale. 2012. “DARPA Mentor Award to Bring Making to Education.” Make Magazine blog entry. http://blog.makezine.com/2012/01/19/darpa-mentor-award-to-bring-making-to-education/.

Doyle, Arthur Conan. 1894. *The Memoirs of Sherlock Holmes.* First. Sherlock Holmes. London: George Newnes. http://www.s4ulanguages.com/holmes1.html.

Drummond, Katie. 2012. “Failure to Launch: Darpa’s Drone Contest Ends Unconquered.” WIRED magazine article. http://www.wired.com/2012/06/darpa-uavforge/.

DuCray, Nathan. 2014. “FAA Posts Do’s and Don’t of Model Aircraft.” Blog entry. http://diydrones.com/profiles/blogs/faa-posts-do-s-and-don-t-of-model-aircraft.

Dutton, William, ed. 1999. *Society on the Line: Information Politics in the Digital Age.* Oxford: Oxford University Press.

Edgar, Andrew. 2006. “Reification.” Encyclopedia entry. In *The Blackwell Dictionary of Modern Social Thought*, ed. William Outhwaite, 563–564. Second. Oxford: Blackwell.

editors, ZIP. 2012. “Non Sequitur News.” *ZIP - Zealous Insanity Parade* 3: Zx0094. http://zine.noisebridge.net/wp-content/uploads/2012/07/zx0094%C6%92_web1.jpg.

Einsiedel, Edna, Erling Jelsøe, and Thomas Breck. 2001. “Publics at the Technology Table: The Consensus Conference in Denmark, Canada, and Australia.” *Public Understanding of Science* 10: 83–93.

Elberse, Anita. 2008. “Should You Invest in the Long Tail?” *Harvard Business Review* (July-August). http://www.geminis.ufscar.br/wp-content/uploads/2009/05/shouldyouinvestinthelongtail.pdf.

Elegin. 2013. “DEFCON 22 Badge Contest: Winning Team = [MLF]: CK(Crypt Killer), Decrement, Elegin, Beaker, Jabroni, Llamaprincess.” Walk Through page. http://elegin.com/dc22/.

Ellul, Jacques. 1964. *The Technological Society.* New York: Vintage Books.

———. 1980. *The Technological System.* New York: Continuum.

Ensmenger, Nathan L. 2010. *The Computer Boys Take over: Computers, Programmers, and the Politics of Technical Expertise.* Cambridge, MA: MIT Press. http://libgen.in/book/index.php?md5=8642cd1d216b5e3aaaf589d009d61ba6.

Epstein, Steven. 1996. *Impure Science: AIDS, Activism, and the Politics of Knowledge.* Berkeley, CA: University of California Press.

Escudero-Pascual, Alberto, Stephane Koch, and George Danezis. 2003. “The Physical Access Security to WSIS: A Privacy Threat for the Participants.” Press release. http://www.nodo50.org/wsis/.

European Commission. 2010. “Commission Welcomes New EU Standards for Common Mobile Phone Charger.” Press release of the European Commission. http://europa.eu/rapid/press-release_IP-10-1776_en.htm?locale=en.

Ezrahi, Yaron. 1990. *The Descent of Icarus: Science and the Transformation of Contemporary Democracy.* Cambridge, MA: Harvard University Press.

Fabio, Adam. 2014. “Beams of Light: An Oscilloscope Demo.” Blog entry. http://hackaday.com/2014/06/27/beams-of-light-an-oscilloscope-demo/.

Farr, Nick. 2009. “Respect the Past, Examine the Present, Build the Future.” blog entry. http://blog.hackerspaces.org/2009/08/25/respect-the-past-examine-the-present-build-the-future/.

Feenberg, Andrew. 1992. “Subversive Rationalisation: Technology, Power and Democracy.” *Inquiry* 35 (3). http://www.sfu.ca/~andrewf/Subinq.htm.

———. 2002. *Transforming Technology: A Critical Theory Revisited.* Oxford: Oxford University Press.

———. 2005. “Critical Theory of Technology: An Overview.” *Tailoring Biotechnologies* 1 (1) (Winter): 47–64.

Ferrer, Mercé Molist. 2014. *Hackstory.es: La Historia Nunca Contada Del Underground Hacker En La Península Ibérica.* Barcelona: Self-published. http://hackstory.es/.

Ferrier, Fran, ed. 2006. *A Profile of Men’s Sheds in Australia: Patterns, Purposes, Profiles and Experiences of Participants: Some Implications for VET & ACE About Engaging Older Men.* Wollongong, NSW: AVETRA; AVETRA.

Fleming, Peter. 2009. *Authenticity and the Cultural Politics of Work: New Forms of Informal Control.* Oxford: Oxford University Press.

Flowers, Stephen. 2008. “Harnessing the Hackers: The Emergence and Exploitation of Outlaw Innovation.” *Research Policy* 37 (2) (March): 177–193. doi:10.1016/j.respol.2007.10.006. http://libgen.in/scimag/index.php?s=10.1016/j.respol.2007.10.006.

Flyvbjerg, Bent. 2006. “Five Misunderstandings About Case-Study Research.” *Qualitative Inquiry* 12 (2) (April): 219–245.

Foster, John Bellamy. 2010. “The Financialization of Accumulation.” *Monthly Review: An Independent Socialist Magazine* 62 (5) (October). http://monthlyreview.org/2010/10/01/the-financialization-of-accumulation/.

Foucault, Michel. 2010. *The Birth of Biopolitics: Lectures at the Collége de France, 1978-1979.* London: Picador.

Foundation, The Linux. 2012. “How Linux Is Built.” Youtube video. https://www.youtube.com/watch?v=yVpbFMhOAwE&feature=youtu.be.

fpletz. 2012. “Launching Rockets with R0kets.” Video post. https://vimeo.com/34415508.

Franco Berardi a.k.a. Bifo. 2005. “Biopolitics and Connective Mutation.” *Culture Machine* 7. http://www.culturemachine.net/index.php/cm/article/viewArticle/27/34.

Franco Berardi a.k.a. Bifo, Marco Jacquemet, and Gianfranco Vitali. 2009. *Ethereal Shadows: Communications and Power in Contemporary Italy*. New York: Autonomedia.

Franklin, Sarah, Celia Lury, and Jackey Stacey. *Global Nature, Global Culture*. New York: Sage.

Frayssé, Olivier. 2013. “Introductory Remarks: Exporting United States Work Models: Laying Out the Issue.” Introductory remarks at the conference ICT and Work: The United States at the Origin of the Dissemination of Digital Capitalism. http://ictandwork.blogspot.hu/2013/04/programme_11.html.

Fukuyama, Francis. 2012. “Surveillance Drones, Take Two.” News article. http://www.the-american-interest.com/2012/09/20/surveillance-drones-take-two/.

Fuller, Matthew. 2003. *Behind the Blip: Essays on the Culture of Software*. New York: Autonomedia.

———. 2005. *Media Ecologies: Materialist Emergences in Art and Technoculture*. Cambridge, MA: MIT Press.

Furr, Nathan, Jeff Dyer, and Clayton M. Christensen. 2014. *The Innovator’s Method: Bringing the Lean Start-up into Your Organization*. Boston, MA: Harvard Business Review Press.

Galison, Peter. 1997. *Image and Logic: A Material Culture of Microphysics*. Chicago, IL: University of Chicago Press.

Garry, Misan, Haren Matt, and Ledo Vicki. 2008. *Men’s Sheds: A Strategy to Improve Men’s Health*. Parramatta, NSW: Mensheds Australia Ltd. http://202.74.67.49/docs/publications/860_Mensheds_Report_Misan.pdf.

Geertz, Clifford. 1973. “Deep Play: Notes on the Balinese Cockfight.” In *The Interpretation of Cultures*. New York: Basic Books.

Geronimo. 2012. *Fire and Flames: A History of the German Autonomist Movement Paperback*. Oakland, CA: PM Press.

Gershenfeld, Neil A. 2005. *Fab: The Coming Revolution on Your Desktop - from Personal Computers to Personal Fabrication*. New York: Basic Books.

Gets Owned), EGO (Everybody. 2012. “EGO[0] Zine.” pastebin phile. http://pastebin.com/NnJ19iPz.

Giddens, Anthony. 1991. *Modernity and Self-Identity*. Cambridge: Polity Press.

Gijzemijter, Martin. 2014. “Dutch Authorities Now Allowed to Film Citizens Using Drones: The Dutch Parliament Has Voted in Favour of Legislation That Will Allow Drone Surveillance Where Public Safety

Is at Risk." New article on ZDNet. http://www.zdnet.com/article/dutch-authorities-now-allowed-to-film-citizens-using-drones/.

Gillette, Halbert Powers, and Richard T. Dana. 1909. *Construction Cost Keeping and Management*. Chicago, IL: Gillette Publishing Company.

Giroux, Henry A. 2007. *The University in Chains: Confronting the Military-Industrial-Academic Complex*. Boulder, CO: Paradigm.

Goddard, John, and Andrew Gillespie. 1986. "The Impact of New Information Technology on Urban and Regional Structure in Europe." *Land Development Studies* (3): 11–32. doi:10.1080/02640828608723897.

Goddard, Michael. 2011. "Towards an Archaeology of Media Ecologies: 'Media Ecology', Political Subjectivation and Free Radios." *The Fibreculture Journal* (17). http://seventeen.fibreculturejournal.org/fcj-114-towards-an-archaeology-of-media-ecologies-%e2%80%98media-ecology%e2%80%99-political-subjectivation-and-free-radios/.

Golding, Barry. 2011. "Shedding Ideas About Older Men's Learning." *Lifelong Learning in Europe* 16 (2) (December): 119–124. http://www.lline.fi/en/News/02122014/shedding-ideas-about-older-men-s-learning.

Golding, Barry, Michael Brown, Annette Foley, Jack Harvey, Lynne Gleeson, and others. 2007. *Men's Sheds in Australia: Learning Through Community Contexts*. Adelaide, SA: National Centre for Vocational Education Research Adelaide, South Australia. http://www.dlc.riversideinnovationcentre.co.uk/wp-content/uploads/2012/10/2007-Mens-sheds-in-Australia-Learning-through-community-contexts.pdf.

Gomez, Ricardo, and Elisabeth Gould. 2012. "Perceptions of Trust: Safety, Credibility and 'Cool'." In *Libraries, Telecenters, Cyberafes and Public Access to ICT: Interntaional Comparisons*, ed. Ricardo Gomez, 1–10. Hershey, PA: IGI Global.

Gomez, Ricardo, ed. 2012. *Libraries, Telecenters, Cyberafes and Public Access to ICT: Interntaional Comparisons*. Hershey, PA: IGI Global.

Gordon, Michael. 2014. "Hacking Canon Pixma Printers: Doomed Encryption." Blog post. http://www.contextis.com/resources/blog/hacking-canon-pixma-printers-doomed-encryption/.

Gray, Mary L. 2013. "Digital Piecework: Lessons from an Ethnographic Study on Amazons MTURK Program." Presentation at the conference ICT and Work: The United States at the Origin of the Dissemination of Digital Capitalism. http://ictandwork.blogspot.hu/2013/04/programme_11.html.

Green, Judith, and Laura Hart. 1999. "The Impact of Context on Data." In *Developing Focus Group Research: Politics, Theory and Practice*, ed. Jenny Kitzinger and Rosaline S. Barbour, 21–35. Thousand Oaks, CA: SAGE.

Greenwalt, Bill, and Alex Pratt. 1998. “Hearings Announced on Computer Security Failures in Government.” Press release of the US Senate. http://web.archive.org/web/20110927215809/http://hsgac.senate.gov/l0pht.htm.

Grenzfurthner, Johannes, and Frank Apunkt Schneider. 2009. “Hacking the Spaces.” Statement on the website of monochrom. http://www.monochrom.at/hacking-the-spaces/.

Gurstein, Michael. 2011. “Telecentres Are Not ‘Sustainable’: Get over It!” Blog entry. https://gurstein.wordpress.com/2011/05/18/telecentres-or-community-access-centres-or-public-interest-access-centres-or-community-technology-centres-etc-
E2%80%9Csustainable%E2%80%9D-get-over-it/.

H.A.C.K. contributors. 2011. “Run an LED Off of a Single Battery!” Wiki page. http://hsbp.org/joulethiefkit-eng&highlight=thief.

Habermas, Jürgen. “Rethinking the Public Sphere: A Contribution to the Critique of Actually Existing Democracy.” *Social Text*.

Haché, Alex. 2014. “Souveraineté Technologique.” In *Souveraineté Technologique*, ed. Alex Haché, 9–17. Dossier Ritimo. Paris: Ritimo. http://www.plateforme-echange.org/spip.php?article102.

HackerspaceSG. 2014. “Frequently Asked Questions.” Webpage. http://hackerspace.sg/faq/.

Halleck, Dee Dee. 1998. “The Grassroots Media of Paper Tiger Television and the Deep Dish Satellite Network.” *Crash Media* (2).

———. 2003. “Indymedia: Building an International Activist Internet Network.” *Media Development* 50 (4): 11–15.

Hamm, Marion. 2003. “A R/c Tivism in Physical and Virtual Spaces.” *Republicart* (9) (September).

Hardt, Michael. 1996. “Introduction: Laboratory Italy.” In *Radical Thought in Italy: A Potential Politics*, ed. Michael Hardt and Paolo Virno, 1–10. Minnesota: University of Minnesota Press.

Harvey, David. 2005. *A Brief History of Neoliberalism.* Oxford: Oxford University Press.

Hayek, Friedrich. 1938. “Economics and Knowledge.” *Economica* 4 (13): 33–54.

Health, Department of, and Ageing. 2010. *National Male Health Policy: Building on the Strength of Australian Males.* Canberra, SA.: Department of Health; Ageing.

Heidegger, Martin. 1993. “The Question Concerning Technology.” In *Martin Heidegger: Basic Writings from “Being and Time” (1927) to “the Task of Thinking” (1964)*, ed. David Farell Krell. San Francisco: HarperCollins.

Heikkilä, Ville-Matias. 2010. "The Future of Demo Art: The Demoscene in the 2010s." Online research paper. http://www.pelulamu.net/countercomplex/the_future_of_demo_art/.

Hempel, Leon, and Eric Töpfer. 2004. "CCTV in Europe: Final Report." Urban Eye research project report. http://www.urbaneye.net/results/ue_wp15.pdf.

Hess, David J. 2005. "Technology- and Product-Oriented Movements: Approximating Social Movement Studies and STS." *Science, Technology and Human Values* 30 (4): 515–535. doi:10.1177/0162243905276499. http://www.davidjhess.org/TPMsFinal.pdf.

HighWiz et al. 2011. "The Big Show: Unofficial Defcon FAQv3." Frequently Asked Questions page. http://defcon.stotan.org/faq/convention3o.htm#ques7.

Himanen, Pekka. 2001. *The Hacker Ethic.* New York, NY: Random House.

Hofman, Wilco Baan. 2013. "Alarm System Security." Research paper.

Hofmann, Jeanette. 1999. "Writers, Texts, and Writing Acts: Gendered User Images in Word Processing Software." In *The Social Shaping of Technology*, ed. Donald MacKenzie and Judy Wajcman, 222–243. Buckingham: Open University Press.

Hornet, CRTC, and Desire. 2015. "8088 MPH." Demo, 1st place at the Revision 2015 demo party (Saarbrücken, Germany) in the "oldskool" category. http://www.pouet.net/prod.php?which=65371.

Hronesova, Jessie, Tristan Caulfield, and Petra Guasti. 2014. "The Xanadu of Surveillance: Report on Security Perceptions in the British Online Media." SECONOMICS ("Socio-Economics meets Security") project research report. http://www.soc.cas.cz/sites/default/files/soubory/the_xanadu_of_surveillance_report_on_security_perceptions_in_the_british_online_media.pdf.

Huang, Andrew "bunnie". 2013. *Hacking the Xbox: An Introduction to Reverse Engineering.* Unlimited Edition. San Francisco, CA: No Starch Press. http://bunniefoo.com/nostarch/HackingTheXbox_Free.pdf.

Hudson, Audrey. 2003. "Bug Devices Track Officials at Summit." News article. http://www.washingtontimes.com/news/2003/dec/14/20031214-011754-1280r/?page=all.

Hughes, Thomas P. 2012. "The Evolution of Large Technological Systems." In *The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology*, ed. Wiebe E. Bijker, Thomas P. Hughes, Trevor Pinch, and Deborah G. Douglas, 51–82. Anniversary edition. Cambridge, MA: MIT Press. http://libgen.in/book/index.php?md5=9DBC0F9472CB052741A982F5FAB8F91B.

Humble, Jez, Joanne Molesky, and Barry O'Reilly. 2015. *Lean Enterprise: How High Performance Organizations Innovate at Scale.* Sebastopol, CA: O'Reilly Media.

Hurst, Nathan. 2014. "TechShop's Not-so-Secret Ingredient: To Achieve Its Prodigious Goals, the Marquee Makerspace Is Dialing in a Precise Method to Make Each New Shop Polished and Profitable." *Maker Magazine* 40 (August-September): 54–58. http://p25ext.lanl.gov/people/hubert/outreach/make/M40_spaces.pdf.

Husserl, Edmund. 1960. *Cartesian Meditations: An Introduction to Phenomenology*. Seventh impression. The Hague, Boston, London: Martinus Nijhoff. http://libgen.in/book/index.php?md5=27BC4CF4CE009C7CF74CAB3BC3390E75.

Igoe, Tom, and Dan O'Sullivan. 2004. *Physical Computing: Sensing and Controlling the Physical World with Computers*. London: Premier Press.

Inter-Corporate Computer & Network Services, Inc. 2012. "Standard Humour." Web page. http://www.openrfc.org/humour.pl.

Isenberg, David. 1997. "Rise of the Stupid Network." Memo. http://www.hyperorg.com/misc/stupidnet.html.

Jameson, Fredric. 1991. *Postmodernism, or, the Cultural Logic of Late Capitalism*. Durham, NC: Duke University Press.

———. 2003. "Fear and Loathing in Globalization: Reflections on William Gibson's Pattern Recognition: A Contemporary Dialectic of Style, as the Verne of Cyberspace Turns to the Branded Present and Its Nauseas." *New Left Review* 23 (September-October). http://newleftreview.org/II/23/fredric-jameson-fear-and-loathing-in-globalization.

Jeff Keyzer, Andie Nordgren, Mitch Altman. 2011. *Soldering Is Easy*. Self-published.

Kaczynski, Theodore John. 1995. "Industrial Society and Its Future." Manifesto, newspaper article in The New York Times and The Washington Post. http://cyber.eserver.org/unabom.txt.

Kamer, Tweede. 2013. "Wetsvoorstel 33582: Wijziging van de Gemeentewet in Verband Met de Verruiming van de Bevoegdheid van de Burgemeester Tot de Inzet van Cameratoezicht." Law amendment by Dutch Parliament. http://www.tweedekamer.nl/kamerstukken/wetsvoorstellen/detail?id=2013Z05461.

Katsiaficas, George. 1997. *The Subversion of Politics: European Autonomous Social Movements and the Decolonization of Everyday Life*. New Jersey, NJ: Humanities Press. http://libgen.in/book/index.php?md5=BC6F41D4F4F8220CF816F03443A92587.

Keddie, Vince. 2006. "Case Study Method." In *The SAGE Dictionary of Social Research Methods*, ed. Victor Jupp, 20–21. London; Thousand Oaks, CA; New Delhi: SAGE.

Kelty, Christopher. 2013. "There Is No Free Software." *Journal of Peer Production* (3) (July). http://peerproduction.net/issues/issue-3-free-software-epistemics/debate/there-is-no-free-software/.

Kelty, Christopher M. 2008. *Two Bits: The Cultural Significance of Free Software*. Durham, NC: Duke University Press. http://twobits.net/.

Kera, Denisa. 2014. “Innovation Regimes Based on Collaborative and Global Tinkering: Synthetic Biology and Nanotechnology in the Hackerspaces.” *Technology in Society* 37: 28–37.

Kera, Denisa, Jan Rod, and Radka Peterova. 2013. “Post-Apocalyptic Citizenship and Humanitarian Hardware.” In *Nuclear Disaster at Fukushima Daiichi: Social, Political and Environmental Issues*, 97–115. Routledge.

Keyzer, Jeff. 2011. “Introducing the MightyOhm Geiger Counter Kit.” Blog entry. http://mightyohm.com/blog/2011/08/geiger-counter-kit-for-chaos-camp-2011-done/.

Kirkpatrick, Graeme. 2008. *Technology and Social Power*. Basingstoke: Palgrave Macmillan.

Kitzinger, Jenny, and Rosaline S. Barbour. 1999. “Introduction: The Challenge and Promise of Focus Groups.” In *Developing Focus Group Research: Politics, Theory and Practice*, ed. Jenny Kitzinger and Rosaline S. Barbour, 1–20. Thousand Oaks, CA: SAGE.

Kleif, Tine, and Wendy Faulkner. 2003. “‘I’m No Athlete [but] I Can Make This Thing Dance!’: Men’s Pleasures in Technology.” *Science, Technology and Human Values* 28 (2) (Spring): 296–325. doi:10.1177/0162243902250908. http://www.studioincite.com/PNT/athlete_dance.pdf.

Klein, Peter G. 2013. “Coase and the Austrians.” Blog entry for the Mises Institute. http://bastiat.mises.org/2013/09/coase-and-the-austrians/.

Kleiner, Dmytri. 2010a. *The Telekommunist Manifesto*. Network Notebooks 3. Amsterdam: Institute for Network Cultures. http://www.networkcultures.org/_uploads/%2333notebook_telekommunist.pdf.

———. 2010b. “Critique of Peer Production Ideology.” P2P Foundation wiki page. http://p2pfoundation.net/Dmytri_Kleiner's_Critique_of_Peer_Production_Ideology.

Knuth, Donald. 1989. “Theory and Practice.” Conference address. http://www-cs-faculty.stanford.edu/~knuth/preprints.html.

———. 1997. *The Art of Computer Programming: Fundamental Algorithms*. 3rd ed. Vol. 1. Boston: Addison-Wesley.

———. 2011. *The Art of Computer Programming: Combinatorial Algorithms*. Vol. 4A. Boston: Addison-Wesley.

Kohtala, Cindy, and Camille Bosqué. 2014. “The Story of MIT-Fablab Norway: A Narrative on Infrastructuring Peer Production.” *Journal of Peer Production* (5) (July).

Kopytoff, Igor. 1986. “The Cultural Biography of Things: Commoditization as Process.” In *The Social Life of Things: Commodities in Cultural Perspective*, ed. Arjun Appadurai. Cambridge: Cambridge University Press.

Kosner, Anthony Wing. 2015. "IndieBio Will Accelerate Synthetic Biology to Tech Startup Speed." News article on Forbes. http://www.forbes.com/sites/anthonykosner/2015/02/20/indiebio-will-accelerate-synthetic-biology-to-tech-startup-speed/2/.

Kostakis, Vasilis. 2013. "At the Turning Point of the Current Techno-Economic Paradigm: Commons-Based Peer Production, Desktop Manufacturing and the Role of Civil Society in the Perezian Framework." *TripleC* 11 (1): 173–90. http://triple-c.at/index.php/tripleC/article/view/463.

———. 2014. "Production and Governance in Hackerspaces: A Manifestation of Commons-Based Peer Production in the Physical Realm?" *International Journal of Cultural Studies* 17 (2).

Kostakis, Vasilis, Michail Fountouklis, and Wolfgang Drechsler. 2013. "Peer Production and Desktop Manufacturing: The Case of the Helix$_T$ Wind Turbine Project." *Science, Technology & Human Values* 38 (6) (November): 773–800. doi:10.1177/0162243913493676. http://libgen.in/scimag/index.php?s=10.1177/0162243913493676.

Kostakis, Vasilis, Vasilis Niaros, and Christos Giotitsas. 2014. "Production and Governance in Hackerspaces: A Manifestation of Commons-Based Peer Production in the Physical Realm?" *International Journal of Cultural Studies* (February). doi:10.1177/1367877913519310. http://libgen.in/scimag/?s=10.1177/1367877913519310.

Lanzeni, Debora. 2015. "Smart Global Futures: Designing Affordable Materialities for a Better Life." In *Designing Digital Materialities*, ed. Sarah Pink, Elisenda Ardevol, and Debora Lanzeni. London: Bloomsbury.

Lash, Scott. 2007. "Power After Hegemony: Cultural Studies in Mutation?" *Theory, Culture, and Society* 24 (3): 55–78.

Latour, Bruno. 1988a. *The Pasteurization of France.* Cambridge, MA: Harvard University Press.

———. 1988b. "Mixing Humans with Non-Humans: Sociology of a Door-Closer." *Social Problems* 35: 298–310.

———. 1991. "Technology Is Society Made Durable." In *A Sociology of Monsters: Essays on Power, Technology and Domination*, ed. Wiebe E. Bijker and John Law, 225–258. London: Routledge.

———. 1992. "Where Are the Missing Masses? The Sociology of a Few Mundane Artifacts." In *Shaping Technology / Building Society: Studies in Sociotechnical Change*, ed. Wiebe Eco Bijker and John Law, 225–258. Cambridge, MA: MIT Press.

———. 1996. *ARAMIS of the Love of Technology.* Cambridge, MA; London: Harvard University Press.

———. 2005. *Reassembling the Social.* Oxford: Oxford University Press.

Law, John. 1991. *A Sociology of Monsters: Essays on Power, Technology and Domination.* New York; London: Routledge.

Levinson, Charles. 2014. “Comey: FBI ‘Grappling’ with Hiring Policy Concerning Marijuana.” *The Wall Street Journal* (May). http://blogs.wsj.com/law/2014/05/20/director-comey-fbi-grappling-with-hiring-policy-concerning-marijuana/.

Levy, Steven. 1984. *Hackers: Heroes of the Computer Revolution.* Anchor Press, Doubleday.

Lialina, Olia. 2012. “Turing Complete User.” Personal website. http://contemporary-home-computing.org/turing-complete-user/.

Lindsay, Christina. 2003. “From the Shadows: Users as Designers, Producers, Marketers, Distributors, and Technical Support.” In *How Users Matter: The Co-Construction of Users and Technology*, ed. Nelly Oudshoorn and Trevor Pinch, 29–50. Cambridge, MA: MIT Press.

Lindtner, Silvia. 2014. “Hackerspaces and the Internet of Things in China: How Makers Are Reinventing Industrial Production, Innovation, and the Self.” *China Information* 28 (2): 145–167.

Lindtner, Silvia, and David Li. 2012. “Created in China. the Makings of China’s Hackerspace Community.” *Interactions* XIX (6) (November-December): 18–22. doi:10.1145/2377783.2377789. http://www.silvialindtner.com/s/p18-created-in-china.pdf.

Lindtner, Silvia, Garnet Hertz, and Paul Dourish. 2014. “Emerging Sites of HCI Innovation: Hackerspaces, Hardware Startups & Incubators.” *ACM Transactions on Computer-Human Interaction.* http://www.wiwi.uni-siegen.de/inno/pdf/lindtner_makerchinainnovation_chi14.pdf.

“Linux Kernel Development: How Fast It Is Going, Who Is Doing It, What They Are Doing, and Who Is Sponsoring It.” 2012. White paper. http://storage.pardot.com/6342/48856/lf_kernel_development_2012.pdf.

Liu, Alan. 2004. *The Laws of Cool.* Chicago, IL: University of Chicago Press.

Local/AFP, The. 2014. “Barcelona Backs down over Squat Demolition.” News article. http://www.thelocal.es/20140530/barcelona-town-hall-backs-down-over-protests.

London Hackspace contributors. 2014. “Door Control System.” Wiki page revision as of 17:49, 17 December 2014 by PeterMeadows. https://hackspace.org.uk/w/index.php?title=Door_control_system&oldid=44505.

Lotringer, Sylvére, and Christian Marazzi. 2007. *Autonomia: Post-Political Politics.* Los Angeles: Semiotext(a).

Lukács György. 1971. *Történelem És Osztálytudat [History and Class Consciousness].* Budapest: Magvető.

Malinowski, Bronisław. 1922. *Argonauts of the Western Pacific: An Account of Nature Enterprise and Adventure in the Archipelagos of Meanesian New Guinea.* London: Routledge & Kegan Paul.

Marino, Mark C. 2006. “Critical Code Studies.” *Electronic Book Review* (December). http://www.electronicbookreview.com/thread/electropoetics/codology.

Markoff, John. 2005. *What the Dormouse Said: How the Sixties Counter Culture Shaped the Personal Computer Industry.* London: Penguin. http://libgen.in/book/index.php?md5=631E8A3A67AB0EA19ECD3DB0E689ADA1.

Marx, Karl. 1845. “The German Ideology: Critique of Modern German Philosophy According to Its Representatives Feuerbach, B. Bauer and Stirner, and of German Socialism According to Its Various Prophets.” https://www.marxists.org/archive/marx/works/1845/german-ideology/index.htm.

———. 2007. *Capital: A Critique of Political Economy.* Vol. Volume I., Part II. New York: Cosimo Classics.

Maxigas. 2012a. “Hacklabs and Hackerspaces — Tracing Two Genealogies.” *Journal of Peer Production* 2. http://peerproduction.net/issues/issue-2/peer-reviewed-papers/hacklabs-and-hackerspaces/.

———. 2012b. “This Is R0ket Science!: Modernity, Capitalism, Liberalism and Hacker Culture.” Masters’ Thesis in Sociology and Social Anthropology at the Central European University, Budapest. http://www.etd.ceu.hu/2012/dunajcsik_peter.pdf.

———. 2012c. “History of Hacklabs & Hackerspaces.” Presentation at the Iberian Hackmeeting in Calafou. http://sindominio.net/hackmeeting/index.php?title=2012/Nodos/History_of_Hacklabs_%26_Hackerspaces.

———. 2014a. “Closure and Stabilization in Open Source Artefacts.” Presentation at the annual 4S (Society for the Social Study of Science) conference, Buenos Aires. http://4sonline.org/meeting.

———. 2014b. “Cultural Stratigraphy: A Historical Rift in the Hacker Scene Between Hacklabs and Hackerspaces.” *Journal of Peer Production* (5) (July).

———. 2014c. “Hackerspaces: The Infrastructure of Open Hardware Production.” Workshop at Backbone409, Calafou. http://backbone409.calafou.org/.

———. 2015a. “Hacklabs and Squats: Engineering Counter-Cultures in Autonomous Spaces.” In *Making Room: Cultural Production in Occupied Spaces*, ed. Alan Moore and Alan Smart. Los Angeles, CA: Other Forms & the Journal of Aesthetics and Protest.

———. 2015b. “Hackers Against Technology: The Politics of Non-Adoption.” Unpublished manuscript.

May First / People Link. 2012. “FBI Seizes Server in Attack on Anonymous Speech.” Press release. https://mayfirst.org/fbi-attacks-anonymous-speech.

McIlroy, Douglas. 1969. “Mass Produced Soptware Components.” In *Software Engineering, Report on a Conference Sponsored by the NATO Science Committee, Garmisch, Germany, 7th to 11th October 1968*, ed. P. Naur and B. Randell, 138–155. Brussels: NATO Science Committee. http://www.cs.dartmouth.edu/~doug/components.txt.

McLuhan, Marshall. 1964. *Understanding Media: The Extensions of Man.* New York: McGraw-Hill.

Merton, Robert K. 1968. *Social Theory and Social Structure.* New York: Free Press.

mh et al. 2011. "R0ket: Concept." Wiki page. http://r0ket.de/concept.

MLF. 2012. "DEFCON 21 Badge Contest: Smashed Again by CK(Crypt Killer), Decrement, Elegin, Beeker." Walk Through page. http://elegin.com/dc21/.

Moilanen, Jarkko. 2013. "Emerging Commons Design Economy." Co-Create.

Moilanen, Jarkko aka kyb3R. 2010. "Hackerspaces, Members and Involvement (Survey Study)." Blog entry. http://extreme.ajatukseni.net/2010/07/19/hackerspaces-members-and-involvement-survey-study/.

———. 2012a. "Good and Bad – Makers and Hackers?" Blog entry.

———. 2012b. "Mapping Hackers: DIY Community Survey 2012 Results." *Journal of Peer Production* (July). http://surveys.peerproduction.net/2012/07/mapping-hackers-diy-community-survey-2012-results/.

Mollick, Ethan. 2005. "The Engine of the Underground: The Elite-Kiddie Divide." *ACM SIGGROUP Bulletin* 25 (2) (February): 23–27. https://dl.acm.org/citation.cfm?doid=1067721.1067726.

Mouffe, Chantal. 2000. *Deliberative Democracy or Agonistic Pluralism.* Vienna: Institute for Advanced Studies.

Mumford, Lewis. 1967. *The Myth of the Machine: Technics and Human Development.* Vol. 1. New York: Brace Jovanovich Harcourt.

Munroe, Randall. 2005. "XKCD: A Webcomic of Romance, Sarcasm, Math, and Language." Web comic. http://xkcd.com/.

Murphy, Michelle. 2006. *Sick Building Syndrome and the Problem of Uncertainty.* Durham, NC: Duke University Press.

Musso, Pierre. 2010. *Saint-Simon, L'industrialisme Contre L'état.* Paris: l'Aube.

Nafus, Dawn. 2012. "'Patches Don't Have Gender': What Is Not Open in Open Source Software'." *New Media & Society* 14 (4): 669–683. https://dawnnafus.files.wordpress.com/2008/09/Fpatches-revised2.pdf.

Noble, David. 1984. *Forces of Production.* New York: Oxford University Press.

Ohlig, Jens, and Lars Weiler. 2007. "Building a Hackerspace." talk at 24C3.

Oldenburg, Ray. 1989. *The Great Good Place.* New York: Paragon Books.

Oost, Ellen Van. 2003. "Materialised Gender: How Shavers Configure the Users' Femininity and Masculinity." In *How Users Matter: The Co-Construction of*

Users and Technology, ed. Nelly Oudshoorn and Trevor Pinch. Cambridge, MA: MIT Press.

Open Source Hardware Association. 2012. "Brief History of Open Source Hardware Organizations and Definitions." Web page. http://www.oshwa.org/research/brief-history-of-open-source-hardware-organizations-and-definitions/.

Oudshoorn, Nelly, and Trevor Pinch, ed. 2003. *How Users Matter: The Co-Construction of Users and Technology*. Cambridge, MA: MIT Press.

Owens, Lynn. 2009. *Cracking Under Pressure: Narrating the Decline of the Amsterdam Squatters' Movement*. Amsterdam: Amsterdam University Press. http://libgen.in/book/index.php?md5=6B76B9E474DD316813CE19CD0B80A6E8.

O'Donnell, Casey. 2014. "Mixed Messages: The Ambiguity of the MOD Chip and Pirate Cultural Production for the Nintendo DS." *New Media & Society* 16 (5) (August): 737–752. doi:10.1177/1461444813489509. http://nms.sagepub.com/content/16/5/737.

O'Mahony, Lorna Fox, David O'Mahony, and Robin Hickey, ed. 2015. *Moral Rhetoric and the Criminalisation of Squatting: Vulnerable Demons?* London: Routledge.

Panzar, John C., and Robert D. Willig. 1981. "Economies of Scope." *The American Economic Review* 71 (2): pp. 268–272. http://www.jstor.org/stable/1815729.

Paoli, Stefano De, and Cristiano Storni. 2011. "Produsage in Hybrid Networks: Sociotechnical Skills in the Case of Arduino." *New Review of Hypermedia and Multimedia* 17 (1) (March): 31–52.

Parikka, Jussi. 2002. *What Is Media Archaeology*. Cambridge: Polity.

Pasteleurs, Frédéric. 2013. "Frédéric Pasteleurs." LinkedIn business-oriented social networking service profile. https://be.linkedin.com/pub/fr%C3%A9d%C3%A9ric-pasteleurs/3b/64b/679.

Pearce, Joshua M. 2012. "Building Research Equipment with Free, Open-Source Hardware." *Science* 337 (6100): 1303–1304. http://www.academia.edu/1933663/Building_Research_Equipment_with_Free_Open-Source_Hardware.

———. 2014. *Open-Source Lab. How to Build Your Own Hardware and Reduce Research Costs*. Amsterdam: Elsevier.

Perens, Bruce. 2007. "Openhardware.org." Web page. http://web.archive.org/web/20071228050204/http://www.openhardware.org/.

Pettis, Bre. 2008. "Building an International Movement: Hackerspaces.org." Talk at 25C3, the 25th Chaos Communication Congress, Berliner Congress Center, 12/27-30.

Pickering, Andrew. 2010. *The Cybernetic Brain: Sketches of Another Future*. Chicago; London: University of Chicago Press.

Pickering, Andrew, and Keith Guzik. 2008. *The Mangle in Practice: Science, Society and Becoming*. Durham, NC: Duke University Press.

Pinch, Trevor J., and Wiebe E. Bijker. 1984. "The Social Construction of Facts and Artefacts: Or How the Sociology of Science and the Sociology of Technology Might Benefit Each Other." *Social Studies of Science* 14 (August): 399–441. http://libgen.in/scimag/get.php?doi=10.1177%2F030631284014003004.

Polgár Tamás. 2005. *Freax: The History of the Computer Demoscene*. Winnenden: CSW-Verlag.

Powell, Alison. 2012. "Democratizing Production Through Open Source Knowledge: From Open Software to Open Hardware." *Media, Culture & Society* 34 (6): 691–708. doi:10.1177/0163443712449497. http://eprints.lse.ac.uk/46173/1/___Libfile_repository_Content_Powell%2C%20A_Powell_%20Democratizing_%20production_%20through_2012_Powell_%20Democratizing_%20production_%20through_2012.pdf.

Pruijt, Hans. 2013. "Culture Wars, Revanchism, Moral Panics and the Creative City. a Reconstruction of a Decline of Tolerant Public Policy: The Case of Dutch Anti-Squatting Legislation." *Urban Studies: An International Journal for Research in Urban Studies* 50 (6) (May): 1114–1129. doi:10.1177/0042098012460732. http://libgen.in/scimag/index.php?s=10.1177/0042098012460732.

Putman, Robert D. 2000. *Bowling Alone: The Collapse and Revival of American Community*. New York, London, Toronto, Sydney, New Delhi: Simon & Schuster.

Putt, Archibald. 2006. *Putt's Law and the Successful Technocrat*. New York: Wiley-IEEE.

Pynchon, Thomas. 2013. *Bleeding Edge*. London: Penguin. http://libgen.in/book/index.php?md5=ada82b7435618060de867dcc6352bc19.

Quintana, David Castells, and Vicente Royuela Mora. 2012. "Unemployment and Long-Run Economic Growth: The Role of Income Inequality and Urbanization." *Investigaciones Regionales* 24: 153–173.

R.E.S.I.S.T.O.R.S. 2009. "History of the R.E.S.I.S.T.O.R.S." Wiki page. http://www.resistors.org/index.php/History_of_the_R.E.S.I.S.T.O.R.S.

r0ket contributors. 2011. "R0ket: This Is R0ket Science!" Git repository hosted on github.com. https://github.com/r0ket/r0ket.

Raison, David. 2010. "Hackerspaces, Postmodern Learning Spheres Beyond the Virtual?" Master's thesis. http://david.raison.lu/hackerspaces_online.pdf.

Randomdata. 2013. "What Is Randomdata?" Web page. http://www.randomdata.nl/blogs/node/53.

Raskin, Max. 2013. "Drone Makers Get Help from the Open-Source, DIY Crowd." Bloomberg Business news article. http://www.bloomberg.com/bw/articles/2013-03-28/drone-makers-get-help-from-the-open-source-diy-crowd.

Raymond, Eric S. 1992. *The New Hacker's Dictionary.* Cambridge, MA: MIT Press.

———. 1999. *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary.* Sebastopol, CA: O'Reilly. http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/.

———. 2000. *Homesteading the Noosphere.* n/a: Thyrsus. http://www.catb.org/esr/writings/homesteading/homesteading/.

———. 2003. *The Art of Unix Programming.* Boston, MA: Addison-Wesley.

Ricoeur, Paul. 1970. *Freud and Philosophy: An Essay on Interpretation.* New Haven, CT: Yale University Press.

Ries, Eric. 2011. *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses.* New York: Crown Business.

Rifkin, Jeremy. 2014. *The Zero Marginal Cost Society: The Internet of Things, the Collaborative Commons, and the Eclipse of Capitalism.* Basingstoke: Palgrave Macmillan.

Rigi, Jakob. 2012. "Peer to Peer Production and Advanced Communism: The Alternative to Capitalism." *Journal of Peer Production* 1. http://peerproduction.net/issues/issue-1/invited-comments/a-new-communist-horizon/.

———. 2013. "Peer Production and Marxian Communism: Contours of a New Emerging Mode of Production." *Capital & Class* 37 (3): 397–416. doi:DOI:10.1177/0309816813503979.

Rivlin, Gary. 2011. "Chain of DIY Stores Sparks Inventions." Newsweek. http://www.newsweek.com/2011/06/26/chain-of-diy-stores-sparks-inventions.html.

Rundle, Guy. 2015. "All Power to the Makerspaces: 3-d Printing in Its Current Form Could Be a Return to 'Small Is Beautiful' Drudgery, but It Has the Potential to Do Much More." *The Jacobin: Reason in Revolt* (17) (April). https://www.jacobinmag.com/2015/04/3d-printing-industrial-revolution-rundle/.

Sale, Kirkpatrick. 1996. *Rebels Against the Future: The Luddites and Their War on the Industrial Revolution: Lessons for the Computer Age.* New York, NY: Basic Books.

sandb. 2011. "Pamela." Software project published on GitHub. https://github.com/sandb/pamela.

Schiller, Dan. 1997. "The Information Commodity: A Preliminary View." In *Cutting Edge: Technology, Information Capitalism and Social Revolution*, ed. Jim Davis, Thomas A. Hirschl, and and Michael Stack, 103–120. London: Verso.

Schneier, Bruce. 2000. *Secrets & Lies: Digital Security in a Networked World.* New York: John Wiley & Sons.

Schrock, Andrew Richard. 2011. “Hackers, Makers and Teachers: A Hackerspace Primer (Part 1).” Blog entry. https://andrewrschrock.wordpress.com/2011/07/27/hackers-makers-and-teachers-a-hackerspace-primer-part-1-of-2/.

Schultze, Thomas, and Almut Gross. 1997. *Die Autonomen: Ursprünge, Entwicklung Und Profil Der Autonomen Bewegung Broschiert [the Autonomists: Origins, Development and Profile of the Autonomous Movement]*. Hamburg: Konkret Literatur Verlag.

Schumacher, E. F. 1973. *Small Is Beautiful: A Study of Economics as If People Mattered*. London: Blond & Briggs. http://libgen.in/book/index.php?md5=5C8427D1C6CF5B18BAE78407D64D87B3.

Sclove, Richard. 2000. “Town Meetings on Technology.” In *Science, Technology and Democracy*, 33–48. Albany, NY: SUNY Press.

Scranton, Philip. 1986. “Beyond Anecdotes and Aggregates: The Pattern of Industrial Decline in Philadelphia Textiles, 1916-1931.” *Antipode* 1 (18): 284–310. http://onlinelibrary.wiley.com/doi/10.1111/j.1467-8330.1986.tb00369.x/abstract?deniedAccessCustomisedMessage=&userIsAuthenticated=false.

Sennett, Richard. 2009. *The Craftsman*. New Haven; London: Yale University Press. http://libgen.in/book/index.php?md5=DB85DF3B46F2E519A111FD5EB961084E.

Seravalli, Anna. 2012. “Infrastructuring for Opening Production, from Participatory Design to Participatory Making?” In *Proceedings of the 12th Participatory Design Conference: Exploratory Papers, Workshop Descriptions, Industry Cases - Volume 2*, 53–56. PDC ’12. New York, NY, USA: ACM. doi:10.1145/2348144.2348161. http://doi.acm.org/10.1145/2348144.2348161.

Shah, Saumil. “15 Years Through Infosec.” Conference presentation slides from NULLCON 2014. http://www.slideshare.net/saumilshah/nullcon2014-12yrs-andabakersdozen.

Shantz, Jeff. 2010. *Constructive Anarchy: Building Infrastructures of Resistance*. Farnham; Burlington, VT: Ashgate. http://libgen.in/book/index.php?md5=cea5f27534977d283748073bbfbda300.

Shea, Robert, and Robert Anton Wilson. 1984. *The Illuminatus! Trilogy*. New York: Dell.

Shirky, Clay. 2003. “Power Laws, Weblogs, and Inequality.” listserv post. http://www.shirky.com/writings/powerlaw_weblog.html.

———. 2008. *Here Comes Everybody: The Power of Organizing Without Organizations*. New York: Penguin Press.

Siefkes, Christian. 2009. “The Tricky Business of ‘Copylefting’ Hardware.” Keinform article. http://keimform.de/2009/the-tricky-business-of-copylefting-hardware/.

———. 2011. “The Emergence of Benefit-Driven Production.” Paper presented at the Open Knowledge Conference, Berlin. http://ceur-ws.org/Vol-739/paper_9.pdf.

Sismondo, Sergio. 2010. *An Introduction to Science and Technology Studies.* Chichester: Wiley-Blackwell.

Slatalla, Michele, and Joshue Quittner. 1995. *Masters of Deception: The Gang That Rules Cyberspace.* New York: HarperCollins.

Smith, Adrian. 2014. "Technology Networks for Socially Useful Production." *Journal of Peer Production* (5).

Smith, Rev. Jeremy. 2013. "The Church as Hackerspace: Breaking Computer Code." Blog entry on Hacking Christianity. http://hackingchristianity.net/2013/10/the-church-as-hackerspace-breaking-computer-code.html.

Söderberg, Johan. 2008. *Hacking Capitalism: The Free and Open Source Software Movement.* London: Routledge.

———. 2010. "Reconstructivism Versus Critical Theory of Technology: Alternative Perspectives on Activism and Institutional Entrepreneurship in the Czech Wireless Community." *Social Epistemology: A Journal of Knowledge, Culture and Policy* 24 (4): 239–262. http://www.tandfonline.com/doi/abs/10.1080/02691728.2010.506962.

———. 2011. "Free Software to Open Hardware: Critical Theory on the Frontiers of Hacking." PhD thesis, Göteborg: Science; Technology Studies, Department of Sociology, University of Gothenburg.

———. 2013. "Determining Social Change: The Role of Technological Determinism in the Collective Action Framing of Hackers." *New Media & Society* 15 (8) (January): 1277–1293. http://nms.sagepub.com/content/15/8/1277.

———. 2014a. "Reproducing Wealth Without Money, One 3D Printer at a Time: The Cunning of Instrumental Reason." *Journal of Peer Production* (4). http://peerproduction.net/issues/issue-4-value-and-currency/peer-reviewed-articles/reproducing-wealth-without-money/.

———. 2014b. "Novelty Against Sovereignity." In *Interference Reader*, ed. Interference. Amsterdam: Interference. http://interference.io/novelty-against-sovereignty.

Söderberg, Johan, and Adel Daoud. 2012. "Atoms Want to Be Free Too! Expanding the Critique of Intellectual Property to Physical Goods." *TripleC: Cognition, Communication, Co-Operation* 10 (1). http://www.triple-c.at/index.php/tripleC/article/view/288.

Söderberg, Johan, and Alessandro Delfanti. 2015. "Hacking Hacked! The Life Cycles of Digital Innovation." *Science, Technology and Human Values.*

Space API authors. 2015. "The Space API: Documentation." Application Programming Interface specification. http://spaceapi.net/documentation.

Spitulnik, Debra. 1994. *Radio Cycles and Recycling in Zambia: Public Words, Popular Critiques, and National Communities.* Ann Arbor, MI: MPublishing, University of Michigan Library. http://quod.lib.umich.edu/p/passages/4761530.0008.007?rgn=main;view=fulltext.

Stajano, Frank. 2003. “Security for Whom? The Shifting Security Assumptions of Pervasive Computing.” *Lecture Notes in Computer Science* 2609: 16–27.

Stallman, Richard. 1999. “On ‘Free Hardware’.” Linux Today article. http://www.linuxtoday.com/infrastructure/1999062200505NWLF.

———. 2010. “The WSIS in Tunis (2005-11-16 to 2005-11-21).” Blog entry. https://www.fsf.org/blogs/rms/entry-20060125.html.

Star, Susan Leigh. 2002. “Infrastructure and Ethnographic Practice: Working on the Fringes.” *Scandinavian Journal of Information Systems* 14 (2): 107–122.

Steele, Guy L., and Eric S. Raymond. 1996. *The New Hacker’s Dictionary.* 3rd ed. Cambridge, MA; London: MIT Press.

Sterling, Bruce. 1992. *The Hacker Crackdown.* New York: Bantam.

Steward, Mark. 2014. “Re: Who Does the RFID Door & Other Machines Access? Knowledge Transfer Request.” Email to the london-hack-space publically archived mailing list. http://comments.gmane.org/gmane.org.hackerspaces.london/44608.

Stoecker, Randy. 1991. “Evaluating and Rethinking the Case Study.” *The Sociological Review* 39 (1) (February): 88–112. http://onlinelibrary.wiley.com/doi/10.1111/j.1467-954X.1991.tb02970.x/abstract.

Swislow, Dan. 2014. “A Permanent Hacker Space in the Brazilian Congress.” Blog entry. http://blog.openingparliament.org/post/72099651071/a-permanent-hacker-space-in-the-brazilian-congress.

Tan, Shuschen. 1995. “Digital City, Amsterdam: An Interview with Marleen Stikker.” *CTheory* (August): a023. http://www.ctheory.net/articles.aspx?id=65.

Tapscott, Don, and Anthony D. Williams. 2006. *Wikinomics: How Mass Collaboration Changes Everything.* New York: Penguin.

Tasajärvi, Lassi. 2004. *Demoscene: The Art of Real-Time.* Helsinki: Even Lake Studios.

Teece, David J. 1980. “Economies of Scope and the Scope of the Enterprise.” *Journal of Economic Behavior & Organization* 1 (3): 223–247. doi:http://dx.doi.org/10.1016/0167-2681(80)90002-5. http://www.sciencedirect.com/science/article/pii/0167268180900025.

Teffer, Peter. 2014. “In Amsterdam, Web Archaeologists Excavate a Digital City: Dutch Researchers Are Trying to Reconstruct a Social-Media Platform from 1994, Raising Questions over How to Preserve Humanity’s Digital Heritage.” News article. http://www.csmonitor.com/World/Europe/2014/0329/In-Amsterdam-web-archaeologists-excavate-a-digital-city.

Terranova, Tiziana. 2000. “Free Labor: Producing Culture for the Digital Economy.” *Social Text* 18 (2): 33–58.

"The Growing Engagement of Emergent Concerned Groups in Political and Economic Life: Lessons from the French Association of Neuromuscular Disease Patients." 2008. *Science, Technology & Human Values* 33: 230–261.

Thoburn, Nicholas. 2003. *Deleuze, Marx and Politics.* Routledge Studies in Social and Political Thought. London; New York: Routledge. http://libgen.in/book/index.php?md5=B67A17C935C3BEBF197FD48B761E08E5.

Thomas, Greg. 2014. "How a German Soda Became Hackers' Fuel of Choice." Article in Vice Magazine. http://www-refresh.vice-motherboard-test.appspot.com/blog/how-a-german-soda-became-hackers-fuel-of-choice.

Thomke, Stefan H., and Eric von Hippel. 2002. "Cutomers as Innovators: A New Way to Create Value." *Harvard Business Review* 80 (4): 74–81. http://www.calt.insead.fr/papers/customers-innovators.pdf.

Thompson, E.P. 1967. "Time, Work-Discipline and Industrial Capitalism." *Past & Present* 38 (1): 56–97.

Thorpe, Charles. 2008. "Political Theory in Science and Technology Studies." In *The Handbook of Science and Technology Studies*, ed. Edward J. Hackett, Olga Amsterdamska, Michael Lynch, and Judy Wajcman. MIT Press.

Torrone, Phillip. 2009. "Open Source Hardware 2009: The Definitive Guide to Open Source Hardware Projects in 2009." Make: Magazine article. http://makezine.com/2009/12/11/open-source-hardware-2009-the-def/.

Toupin, Sophie. 2014. "Feminist, Queer and Trans Hackerspaces: The Crystallization of an Alternate Hacker Culture?" *Journal of Peer Production* (5) (October).

Touraine, Alain. 1969. *La Société Post-Industrielle [the Postindustrial Society].* Paris: Denoel.

Treb0r. 2011. "London Hackspace." Blog entry. http://treb0r.net/2011/10/london-hackspace/.

Trixster. 2015. "8088 MPH: We Break All Your Emulators." Blog entry. http://trixter.oldskool.org/2015/04/07/8088-mph-we-break-all-your-emulators/.

Troxler, Peter. 2010. "Commons-Based Peer-Production of Physical Goods: Is There Room for a Hybrid Innovation Ecology?" Paper presented at the 3rd Free Culture Research Conference, Berlin. http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1692617.

———. 2015. "Fab Labs Forked: A Grassroots Insurgency Inside the Next Industrial Revolution." *Journal of Peer Production* (5) (July). http://peerproduction.net/issues/issue-5-shared-machine-shops/editorial-section/fab-labs-forked-a-grassroots-insurgency-inside-the-next-industrial-revolution/.

Troxler, Peter, and Maxigas. 2014. "We Now Have the Means of Production, but Where Is My Revolution?" *Journal of Peer Production* (5) (October). http:

//peerproduction.net/issues/issue-5-shared-machine-shops/editorial-section/editorial-note-we-now-have-the-means-of-production-but-where-is-my-revolution/.

Turing, Alan. 1950. "Computing Machinery and Intelligence." *Mind* LIX (236) (October): 433–460. doi:doi:10.1093/mind/LIX.236.433. http://orium.homelinux.org/paper/turingai.pdf.

Turner, Fred. 2006. *From Counterculture to Cyberculture: Stewart Brand, the Whole Earth Network, and the Rise of Digital Utopianism.* Chicago, IL: University Of Chicago Press. http://libgen.in/book/index.php?md5=6916B53A2F2766021740909943602E3F2.

Ultratux. 2015. "ACTA/Closing up." TechInc. hackerspace wiki page. https://wiki.techinc.nl/index.php/ACTA/Closing_up.

University of Exeter. 2012. "Violins 'Printed' in Exeter." Press release. http://www.exeter.ac.uk/news/archive/2012/january/title_175937_en.html.

Vandenberghe, Frédéric. 2008. *A Philosophical History of German Sociology.* Critical Realism. London; New York: Routledge.

———. 2013. "Reification: History of the Concept." *Logos: A Journal of Modern Society & Culture* 12 (3) (October). http://logosjournal.com/2013/vandenberghe/.

VileR. 2015. "CGA in 1024 Colors: A New Mode - the Illustrated Guide." Blog entry. http://8088mph.blogspot.com/2015/04/cga-in-1024-colors-new-mode-illustrated.html.

Virno, Paolo. 2004. *A Grammar of the Multitude: For an Analysis of Contemporary Forms of Life.* New York: Semiotext[e]. http://libgen.in/book/index.php?md5=DB0A96229A342A390E8C0CE4BAA42700.

von Hippel, Eric. 1986. "Lead Users: A Source of Novel Product Concepts." *Management Science* 32 (7): 791–805.

———. 2005a. *Democratizing Innovation.* Cambridge, MA; London: MIT Press. http://web.mit.edu/people/evhippel/democ1.htm.

———. 2005b. "Democratising Innovation: The Evolving Phenomenon of User Innovation." *Journal Für Betriebswirtschaft.* http://www.oecd.org/sti/inno/37450155.pdf.

Wainwright, Hilary, and Michel Bauwens. 2012. "Peer-to-Peer Production and the Coming of the Commons." *Red Pepper* (July). http://www.redpepper.org.uk/the-coming-of-the-commons/.

Wallerstein, Immanuel. 2004. *World-Systems Analysis: An Introduction.* Durham; London: Duke University Press.

Warwick, Graham. 2012. "What Darpa Learned from UAVForge." Aviation Week article. http://aviationweek.com/blog/what-darpa-learned-uavforge.

Weber, Max. 1958. *The Protestant Ethic and the Spirit of Capitalism.* New York: Charles Scribner's Sons.

Weber, Steven. 2004. *The Success of Open Source.* Massachusetts, CA: Harvard University Press.

Weinberg, Michael. 2015. "Your Input Needed for Open Source Hardware Certification." Blog entry on the OSHWA website. http://www.oshwa.org/2015/06/02/your-input-needed-for-open-source-hardware-certification/.

Westcott, Matt. 2012. "Wolfy: Wolfenstein for the R0ket Badge." Blog entry. http://matt.west.co.tt/demoscene/wolfy/.

Weyand, Tobias, and Christian Buck. 2015. "Okr0ket: A R0ket Dating App." Lightining talk at the 28th Chaos Communication Congress (28C3) "Behind Enemy Lines", annual meeting of the Chaos Computer Club, Berlin. http://events.ccc.de/congress/2011/wiki/Lightning_Talks.

Wijnen, Bas, G. C. Anzalone, and Joshua M. Pearce. 2014. "Open-Source Mobile Water Quality Testing Platform." *Journal of Water, Sanitation and Hygiene for Development* 4 (3): 532–537. doi:10.2166/washdev.2014.137. https://www.academia.edu/8319858/Open-source_mobile_water_quality_testing_platform.

Wikipedia. 2015. "Electronic Frontier Foundation." http://en.wikipedia.org/w/index.php?title=Electronic_Frontier_Foundation&oldid=644772837.

Wikipedia Contributors. 2015. "Hackmeeting." Encyclopedia entry. https://it.wikipedia.org/w/index.php?title=Hackmeeting&oldid=70305777.

Wikipedia contributors. 2010a. "Hacklab: Revision History." http://en.wikipedia.org/w/index.php?title=Hacklabaction=history.

———. 2010b. "Talk:Hacklab." http://en.wikipedia.org/wiki/Talk:Hacklab.

———. 2014a. "Wikipedia, the Free Encyclopedia: ASCII (Squat)." http://en.wikipedia.org/w/index.php?title=ASCII_(squat)&oldid=540947021.

———. 2014b. "Hackerspace: Revision History." http://en.wikipedia.org/w/index.php?title=Hacklabaction=history.

Willemsen, Merel. 2006. "Telestreet: Squatting Frequencies." *Untitled Magazine* (37) (Spring).

Williamson, Oliver E. 1983. *Markets and Hierarchies: Analysis and Antitrust Implications.* New York: Free Press.

———. 1995. "Transaction Cost Economics and Organization Theory." In *Organisational Theory: From Chester Barnard to the Present and Beyond*, ed. Oliver E. Williamson, 207–257. New York: Oxford University Press.

———. 1998. *The Economic Institutions of Capitalism.* New York: Free Press.

Wilson, Giles. 2005. "Shed Heaven." BBC News Magazine article. http://news.bbc.co.uk/2/hi/uk_news/magazine/4543675.stm.

Wilson, Nathan J., and Reinie Cordier. 2013. "A Narrative Review of Men's Sheds Literature: Reducing Social Isolation and Promoting Men's Health and Well-Being." *Health and Social Care in the Community* 21 (5): 451–463. doi:10.1111/hsc.12019. http://libgen.in/scimag/index.php?s=%2010.1111/hsc.12019.

Wilthagen, Ton, and Frank Tros. 2004. "The Concept of 'Flexicurity': A New Approach to Regulating Employment and Labour Markets." *Transfer: European Review of Labour and Research* 10 (2): 166–186. http://trs.sagepub.com/content/10/2/166.short.

Winner, Langdon. 1999. "Do Artifacts Have Politics?" *Daedalus* 109 (1) (Winter): 121–136. doi:10.2307/20024652. http://libgen.in/scimag/index.php?s=10.2307/20024652.

Winthrop-Young, Geoffrey. 2000. "Silicon Sociology, or, Two Kings on Hegel's Throne? Kittler, Luhmann and the Posthuman Merger of German Media Theory." *Yale Journal of Criticism* 13 (2): 391–420. doi:offrey Winthrop-Young.

Wires, Mike. 2014. "Intro to 3D Printing." Blog entry. http://mikewires.com/intro-to-3d-printing/.

Wolf, Patricia, Peter Troxler, Pierre-Yves Kocher, Julie Harboe, and Urs Gaudenz. 2015. "Sharing Is Sparing: Open Knowledge Sharing in Fab Labs Image." *Journal of Peer Production* (5) (July). http://peerproduction.net/issues/issue-5-shared-machine-shops/peer-reviewed-articles/sharing-is-sparing-open-knowledge-sharing-in-fab-labs/.

Wright, Steven. 2002. *Storming Heaven: Class Composition and Struggle in Italian Autonomist Marxism.* London: Pluto Press.

Wyatt, Sally. 2008. "Challenging the Digital Imperative." Inaugural Lecture. http://depot.knaw.nl/7740/1/inaugural-lecture-28032008.pdf.

Wyatt, Sally, and Brian Balmer. 2007. "Home on the Range: What and Where Is the Middle in Science and Technology Studies?" *Science, Technology & Human Values* 32 (6) (September): 619–626.

Wynne, Brian. 1996. "May the Sheel Safely Graze? A Reflexive View of the Expert- Lay Knowledge Divide." In *Risk, Environment & Modernity*, ed. Scott M. Lash, Bronislaw Szerszynski, and Brian Wynne, 44–83. London: Sage.

———. "Seasick on the Third Wave? Subverting the Hegemony of Propositionalism." *Social Studies of Science* 33: 401–419.

Yin, Robert K. 2002. *Case Study Research: Design and Methods.* 3rd ed. Thousand Oaks, London, New Delhi: Sage.

Yuill, Simon. 2008. "All Problems of Notation Will Be Solved by the Masses." Article in Mute: Politics and Culture after the Net.

Zahra, Fatima Tuz. 2012. "Why Are Telecenters Not Sustainable? Why Is It Important to Learn from Failures?" Blog entry on ICTs for

Bottom of the Pyramid. https://ict4bop.wordpress.com/2012/04/09/why-are-telecenters-not-sustainable-why-is-it-important-to-learn-from-failures/.

Øverenget, Einar. 1998. *Seeing the Self: Heidegger on Subjectivity.* Dordrecht, Boston; London: Kluwer Academic Publishers. http://libgen.in/book/index.php?md5=4e7b92b867a7447692d4e931bab856ce.