This is r0ket science!
Modernity, Capitalism and Liberalism in Hacker Culture

By
Dunajcsik Péter maxigas

Submitted to
Central European University
Department of Sociology and Social Anthropology

In partial fulfillment of the requirements for the degree of Master of Arts

Supervisors:
Jean-Louis Fabiani
Jakob Rigi

Budapest, Hungary
2012

This is r0ket science!
Modernity, Capitalism and Liberalism in Hacker Culture

2011 August, Finowfurt airport, Germany. Around 5000 hackers gather for a week in tents and hangars to celebrate knowledge, sharing and creativity. The conference is called Chaos Communication Camp, and all participants receive a conference badge called the r0ket. It displays your name on an LCD panel, but it does much more than that: it is a primitive computer and wireless device designed to trigger all the cultural allergies of hackers. You can play the famous retro game Space Invaders on it, and the high scores of the game are shared amongst the crowd. Two hours after takeoff the high score system is already hacked: somebody leads the top of the list with -27500 points. Before the end of the conference, the badge is used as a component in a Do It Yourself Geiger counter, as a remote control for drones, as an electronic torch, and a dozen other amazing purposes. It has no price and it cannot be bought, but anybody can build one from basic components following the online documentation. It is a typical result of the work that goes on in more than 500 hackerspaces around the world.

A serious attempt to dissect the relevance of the hacker movement should first and foremost start by looking at practice. This practice becomes intelligible when weighted against the social totality. ~ Söderberg (2008, 181)

Table of Contents

Abstract

In this study I follow a technological artifact called r0ket as it moves through the hackerspace scene. I concentrate in tracing the connections the r0ket makes inside and outside the scene as well as its internal technological structure. Based on the ethnographic data, I ask whether these connections make sense in the framework of categories like modernity, liberalism and capitalism. I posit an interactive relationship between the categories and the network data, in which the data can modify categories, but categories can also highlight the more interesting patterns and connections in the data itself. Finally, I ask if theories of nonmodernity can explain some of the discrepancies between categories and data.

Acknowledgements

The text below is written from the inside, based on my experiences in the hacker scene. When writing from the inside, one chief difficulty is to port the native knowledge to academic knowledge, and to explain evidences in a way that laymen can parse and find useful, especially in the case of writing about technology for social scientists. This is where the collaboration with my supervisors from the CEU, Jakob Rigi — who has first hand knowledge of the field as an ethnographer and the problems of peer production as a sociologist — and Jean-Louis Fabiani — who has expertise in issues of cultural production — proved invaluable. Additionally, discussions with Andrew Pickering have been instrumental in developing the material on cybernetics. Furthermore, in order to ensure that I am not only presenting my personal delusions about the scene, or I am not disclosing anything sensitive, peer review of practitioners was absolutely essential.

Questions

In this study I follow a technological artifact called r0ket as it moves through the hackerspace scene. I concentrate in tracing the connections the r0ket makes inside and outside the scene as well as its internal technological structure. Based on the ethnographic data, I ask whether these connections make sense in the framework of categories like modernity, liberalism and capitalism. I posit an interactive relationship between the categories and the network data, in which the data can modify categories, but categories can also highlight the more interesting patterns and connections in the data itself. Finally, I ask if theories of nonmodernity can explain some of the discrepancies between categories and data.

My approach is more ethnographic and detailed than most previous research and therefore presents an opportunity to test the validity of existing claims about hacker culture against actual practices that abound around the r0ket device. As with all case studies, the prospect of such research is always also to produce not only new data, but a new understanding of the phenomena at hand. One possible direction for expansion is to look beyond the motivation and subjective experiences of hackers by studying their products.

I claim that researching hacker culture through a physical artifact is a ground breaking approach, since most arguments in the existing literature depend on examples and mechanisms of online collaboration. While the r0ket is embedded in a hybrid virtual-actual space, it is a physical artifact that is mostly handled by embodied communities in self-organised spaces. The use cases of the r0ket show that it is primarily used for developing electronic hardware. Therefore this case study can answer the question whether the hacker way of working can be extended to physical artifacts, especially complex ones. Thus results should complement existing research on hacker culture in a productive way.

Methodology

The material presented here has been gathered through three relatively distinct set of research methods. Firstly, I did field work, mostly active participant observation and desktop research in hacker gatherings and hacker spaces where r0ket surfaced, as well as the virtual communication spaces in which r0ket is discussed, documented, and enjoyed. Secondly, I conducted semi-structured interviews with people associated with the r0ket, both offline and online. Thirdly, I undertook technical interrogation of the r0ket device itself which involved taking it apart and putting it together, reprogramming it and soldering extensions on it.

The combination of these methods were inspired by the "fusion methodology" developed at the Citizen Lab in Canada, where they combine (1) "on the ground" field investigation including participant observation and interviews with (2) online interactions and (3) technical analysis (Forlano 2012). They claim that the combination of such methods yields higher validity since findings on one side can be tested and verified on others (Information Warfare Monitor & Shadowserver Foundation 2010, 3). My own experience supports these claims. During my own research, field work and technical interrogation often yielded new questions that could be answered through interviewing, and interview results pointed out new directions for technical interrogation and field work.

Participant observation in embodied communities can be undertaken with the classical methods described by Bernard (1995a), in addition to Hammersley, Martyn and Paul Atkinson (1983). While these guidelines are useful in the case of virtual communication spaces to some extent, they have to be complemented by "natively digital" methods as described by Richard Rogers from the pioneering methodological workshop at the University of Amsterdam (2009). Rogers emphasises the understanding of the technical infrastructure and how it structures relationships, objects and interactions. He contrasts "virtual methods" with "digital methods" — while the former refers to

traditional data collection techniques applied to online environments, the latter means looking at units of analysis that are not found offline. One example from my research is the "commit": a packaged change to software code — here, the firmware of the r0ket — consisting of deleted and added lines of code and the meta-information about who has done the change, why and when ("Graphs of r0ket/r0ket", *Github.com*). This enabled me to see how many people worked on the software parts of the r0ket, in what rhythm, and what was the social dynamics between the developers.

Hackerspaces themselves, as physical environments, support certain kinds of collaboration and make other kinds difficult. Online media like mailing lists powered by Mailman, the Git distributed version control system for software development built into the github.com website, or the soup.io micro blogging service — all used by the r0ket team — each have their supported association models and each tend to produce certain kinds of results better than the other options. They are important data sources and present a variety of different data sets which have to be interpreted contextually.

During the course of the actual research I visited and stayed in hackerspaces in Germany, London and Budapest. The most important research sites, however, were hacker gatherings where the device was distributed and utilised. The two principal ones were the Chaos Communication Camp at the Finowfurt airport in Germany (10-14 August, 2011) where the r0ket was premiered and the Chaos Communication Camp where the r0ket was first sold (also called 28C3; 27-30 December, 2011). These were important for mapping the distribution process and usage patterns of the r0ket. In addition to participant observation, during the latter gathering I interviewed two members of the r0ket development team separately, and then three other team members together. These interviews generally lasted around an hour, and used the semi-structured interview model

described by Bernard (1995b) and Briggs (1983). Most of the material about production came from these interviews. Additionally, I conducted unstructured interviews with two participants at the London and Budapest hackerspaces on the r0ket.

In terms of online research, since the summer of 2011 I have been following the discussion mailing list of hackerspaces.org, the discussion and development list for the r0ket device itself, and the IRC (Internet Relay Chat) rooms associated with both projects. In addition to archiving the discussion in the chat rooms I also used them interactively to clarify my understanding of certain situations and to find out facts that were missing from elsewhere. Both initiatives (hackerspaces.org and the r0ket device) have their associated wiki websites which collect documentation on them. I studied these websites and followed the twitter and soup.io accounts of the r0ket. Both are micro blogging platforms that proved to be the main source for cataloguing the use cases of the device.

Technical interrogation consisted of a number of activities. I gathered and bought both versions of the r0ket that are available at the moment and assembled them based on the bundled instructions. I reviewed the software and hardware elements of the r0ket and tested their functionality, mainly in the context of the hacker gatherings. These experiences are reflected in the chapter presenting the device, especially the sections on hardware and software.

In line with Weber's idea of sociological objectivity where he argues that individuals — even scientists — always have their own bias, so that scientific validity depends on the understanding of the specific perspective from which a piece of research was conducted (Weber 1949), it seems prudent to clarify the situated dimension of this inquiry. My own position in the field is that I have been involved in the hacker scene as a technological activist since 2002 (for a decade now). In 2010 I got involved in the hackerspaces with the foundation of the Hungarian Autonomous Center for Knowledge in Budapest and the Hackney Crack House in London. As an anarchist I

am interested in the political potentialities of hackers — dissatisfied with the restrictive approach of political hackers on the one hand and dissatisfied with the lack of political consciousness of the nonpolitical hackers on the other. This paper reflects this ambiguous engagement with technological politics. Furthermore, as a practitioner I am obviously closer to the subjects and objects of the research than most other scientists. This is reflected in an emphasis on ethnographic detail that is seldom found in the existing literature, but could just as well go too far or too close in some cases. As the reader will discover, the excess of science and the science of excess are also the underlying themes of the paper.

For the above reasons, a specific problem with the current research concerned editorial choices about what material to include and exclude from the presentation of the case. Even if the case is clearly bounded and anchored to an object, I collected a vast amount of data bound together by what I see as close connections across subsets. The decision to describe the case in a separate chapter followed by the analysis was one way to reduce the complexity arising from this situation. I believe that this was the hardest aspect of conducting research on a hybrid (online-offline) field, especially one that I know relatively well from first hand experience.

I have chosen the r0ket device consciously in order to be able to maintain a degree of analytical distance — the third key challenge for more or less native researchers. The r0ket is not a product of my peer group, and I have no vested interest in it other than for the purposes of this research project. Furthermore, as I wrote above, it has been launched in 2011 when I began the research so I have been able to see its development from a research perspective since its inception, without being entangled in its web as a practitioner.

Finally, focusing on an object rather than studying a group of people directly enables the researcher to increasingly respect the privacy of subjects — a value held high in the hacker community. For the same reason, one of the limitations of the study is that the backgrounds of persons which feature in it are not described in detail. At this stage I decided to accept this as the limitation of the field itself and my position in it.

Literature review

This chapter is comprised of three sections. Key concepts and categories like hacking, as well as liberalism, capitalism and modernity are defined based on the sociological literature. Next, the state of the social scientific knowledge on hackers is presented briefly through three or four seminal authors. Finally, a couple of ideas put forward by more philosophically inclined critiques of modernity are reviewed which will be useful for the analysis.

Key concepts: hacking and the liberalism, capitalism, modernity triad

The definition of hacking itself is the subject of much debate and amusement in the community, but it is perhaps most appropriately defined in the definitive Jargon File ("a common heritage of the hacker culture") like this: "Hacking might be characterised as 'an appropriate application of ingenuity'. Whether the result is a quick-and-dirty patchwork job or a carefully crafted work of art, you have to admire the cleverness that went into it." (Raymond et al 1992) Therefore, a hack can be recognised by the particular flavour of intellectual, aesthetic and moral satisfaction or frustration that it brings to the situation. Consequently, hacker is more often than not a compliment or a social status that has to be earned, which means that people are generally frowned upon when they apply it to themselves. In mainstream discourse, it is still refers to

groups or individuals who exploit their expertise to take advantage of their technological skills to penetrate computer networks and wreak havoc for "fun and profit". While some hackers happily engage in similar activities, I hope to show here that hackerdom is primarily a productive activity which covers much more ground than attacks on network infrastructure. Some hackers and journalists make a distinction between hackers and crackers, where the latter are supposed to be the destructive and criminal elements. I strongly oppose distinction because firstly it does not account for the political aspect of attacks against infrastructure, secondly it divides the scene and illegalises some of its participants, and thirdly it does not take into account the large and deep grey areas between legality and illegality. Hackers have an exciting and complex history that resulted in the development of various more or less separate scenes, one of which is the hackerspaces (maxigas, forthcoming).

Firstly, capitalism is a set of social relations understood via their economic aspect as a specific mode of production. As a historical category it superseded feudalism, and now it is in a stage that can be called late capitalism (authors like Adorno, Harvey, Jameson, Wallerstein or Negri all take this position). It is characterised by private ownership of the means of production which aids endless capital accumulation. As an economic system it produces exploitation and alienation. Secondly, liberalism is a political ideology that emphasises individual freedom and equal rights. It is sometimes viewed as a "centrist" ideology between the radical/communist "left" and the conservative/fascist "right". Nation states were mostly set up on liberal grounds, thus the core liberal tenets are often accepted by groups that otherwise occupy very different political positions. David Harvey (2007) argues that in the late capitalist era it mutated into what he and others call neoliberalism. Liberalism and especially neoliberalism is widely criticised for its "economic blindness", meaning that it fails to consider economic factors when evaluating human rights issues such as discrimination. The notion of structural violence (Gilman 1983) is useful in this regard. Finally, modernity is the cultural experience of capitalism. Jameson and Lyotard argue

that the cultural experience of late capitalism is post-modernity, although this thesis is widely debated by Lash, Beck, Giddens, etc. Zygmunt Bauman proposed the term liquid modernity as an alternative, while others sticked to the notion of late modernity in conjunction with the late capitalism mentioned above. I claim that hackers reconfigure these abstract machines because the categories defined under these terms do not work according to their needs and they have better ideas how to take them apart and assemble them in a different way. Reconfiguration is a good word because it expresses the ambiguity of the operation: only history will tell if hackers go beyond modernity/liberalism/capitalism, or whether they are the vanguard of their historical mutation and expansion. Of course, reconfiguration is also something that hackers actually do a lot with their computers.

Liberalism, capitalism and modernity are three concentric circles where the smaller circle is not possible without the bigger circle, and conversely, the bigger circle includes more options than the smaller circle. Therefore I will discuss them in the Analysis chapter starting from the smaller and proceeding towards the larger. Most of the conclusions from these can be drawn together to bear on the cybernetics as it has been put forward by Pickering. Finally, I will pay a small tribute to Latour's networks by showing how the network frame of interpretation makes some of the conclusions more lucid. Alternatively, the triad liberalism-capitalism-modernity can be seen as roughly corresponding to the personal-structural-historical constructions, but naturally the matter is more complicated since each constructs its own subjects, structure and history. However, the arc of my analysis is built on taking liberalism from the point of view of subjectivation, capitalism as a political-economical structure in which these subjects are embedded and modernity as the wider historical context of that process.

Literature on hackers

For the sake of brevity I have chosen a handful of key authors who give enough theoretical tools for the present paper to pursue the research goals. Pekka Himanen's (2001) treatment of the hacker ethic is useful mainly for the discussion of liberalism as an ideology since he traces down a whole set of personal freedoms hackers are supposed to enjoy. Johan Söderberg's (2008) argument about hacking as an anticapitalist labour struggle will inform the inquiry into hackers and capitalism. Eric S. Raymond (1999), who is looking at the free software development methodology from a more technical direction than the other authors, will be instructive when it comes to the idea of modern versus nonmodern scientific practice (explained later in the discussion of Pickering and Latour). Finally, Yochai Benkler's (2011) theory of peer production is an attempt to present a coherent framework for the emerging mode of production posited by the other authors.

There are many reasons to argue that hackers disrupt the logic of liberalism, capitalism and modernity. Himanen, Söderberg and Raymond all agree that in general hackers find satisfaction in their work, so they are not motivated by financial gain or the self-esteem stemming from a hardworking life which characterises the Protestant ethic (Weber 2002). In particular, they are driven by the tripartite motivation of passion, play and caring. All three authors point out that the massive collaboration organised on these grounds presents a theoretical and perhaps practical challenge to the existing concepts of individual self-satisfaction (liberalism), prevailing business models of capital accumulation (capitalism) as well as traditional software development practices and scientific research (modernity).

Benkler, looking at the ecology of peer production (free software but also Wikipedia and other projects), elucidates more carefully than the others how the rise of the Internet facilitates these collaborations, including the distribution of labour between a small circle of core developers, a

wider mass of contributors and an audience of freeriders. He correctly points out — along with the more widely read Lawrence Lessig (2004) — that under such circumstances intellectual property, which is one of the foundations of capitalist accumulation based on immaterial labour, which is in turn the hegemonic form of labour in postfordist economies, is increasingly becoming more of an obstacle than a productive force.

Proceeding from the latter to the former, each author offers an interesting point about these challenges. Firstly, Raymond demonstrates how the free software development methodology — that he captures in the image of the *Bazaar*, counterposed with the Cathedral — can produce programming code of superior quality, addressing the needs of users directly as they emerge. In an epilogue to his analysis he even shows how corporate management limits the productivity of programmers. Secondly, Söderberg theorises that the *"play struggle"* of hackers is a more or less covert labour struggle against capitalist arrangements resulting in alienation (hence "play") and exploitation (hence "struggle"). He points to the social scientific and political significance of hacking as the germ form of unalienated productive activity. Philosophically, the original Marxist anthropology posits the essence of human beings as the exercise of creativity, orientated towards beauty. Taken politically, the unalienated labour forms the economic basis of communism. Thirdly, Himanen points out that the traditional and widely popular emic concept of the *hacker ethic* is a viable and vibrant alternative to the ideology of the Protestant ethic. His ultimate conclusions stem from the analysis of Protestant and hacker subjectivities, or the process of subjectivation itself. He finds that the subjectivities tailored to fit Protestant values and the criteria of modern rationalisation are transformed into cold cyborgs — men become machines: "In the end, the ideals of a network enterprise or person and those of a computer or network are actually the same: the ability to function flexibly in a way optimal for each project goal, while

maintaining stability at high speed." (128) On the other hand, the very hackers who develop an intimate relationship to computers and networks — networked computers — through their work, make machines like man: free, unpredictable, and amazing.

It is plain to see that the basic contradiction these authors face can only be resolved by a dialectician equipped with the theoretical tools of political economy like Söderberg. How is it possible that hackers are mostly happy freelancers or even successful entrepreneurs at the same time their attitudes fall so fundamentally out of line with the hegemonic configurations of liberalism-capitalism-modernity? How is it possible that some of the richest men on the planet like Bill Gates emerged from the hacker scene? How is it possible that their hacks are starred in the media as the sign of the times, fuel the new economy through shooting star start up companies, and are at the forefront of modern computer science as a discipline? How is it possible that the hacker favourite Unix family of operating systems is powering the majority of servers on the Internet (e.g. Linuxen and BSDs), as well as the majority of smart phones (Android), while making advances on the desktop computing front (OS X powering all Macintosh computers, GNU/Linux on Chinese Longsoon machines, Ubuntu Linux for IBM PC compatible machines)?

The most convincing explanation, which Söderberg, Rigi and to a lesser extent Lessig and Benkler proposes is that peer production is a new mode of production which is articulated in the contemporary capitalist context. Peer production contributes to the development of a resource pool available for everybody — called a commons —, which can be exploited by capitalist and non-capitalist actors alike; while the massive, illegal piracy of gated intellectual property actually raises the value of sold copies through indirect market effects similar to the logic of the commons. Therefore the current arrangements are at least partly compatible with the novel developments. Notably, this state of affairs is in stark opposition with the situation of many social movements, whose proposed visions are often directly contradictory to the current systems.

Importantly, there is a strong message of freedom throughout the literature on hacker culture, which is frequently connected — a bit mystically — through play to specifically aesthetic sensibilities. The fifth commandment of the hacker ethic as it is formulated by Steven Levy (2010) states that "You can create art and beauty on a computer." When Linux Torvalds is asked "What do you think is the most important skill every programmer should posses?", he gives his stock answer; "It's a thing I call 'taste'." (Ahmad 2005). Söderberg theorises this mainly through the early Marx, the late Marcuse and the Letters of Schiller; Himanen through the programmers' closeness to God who creates a world through his words; while for Raymond — a programmer himself and the author of *The Art of Unix Programming* (itself a homage to Donald Knuth's epic multivolume *The Art of Computer Programming*) —, this is not even a question. While for me it remains a mystery, from time to time I will come back to the poetic aspects of the r0ket device throughout this paper from both a theoretical and an ethnological point of view.

My own enquiry will test and refine these observations made by the key authors featuring in this chapter. Going a step further, my ethnographic research addresses a question that all the three principal authors explicitly raise as a promising direction for future inquiries. In the words of Himanen, "It remains to be seen what great things individuals' direct cooperation will accomplish in our 'flesh reality'" (2008, 81). The first step is to look at what happens when hackers come together in the flesh to create and play with physical artifacts. Does hardware hacking change the patterns of cooperation described by the authors above, who are focusing on software projects? How does peer production can work in the context of embodied communities and, most importantly, material goods?

Literature on alternatives to modernity

While the previously described authors already present alternatives to attitudes that can be called modern, I argue that their questions are best taken up when supplemented by the observations of others who explicitly thematise the issue of modernity versus nonmodernity. Such authors — who I treat in this section — tend to base their theories on a dichotomy that I seek to deconstruct in my ethnography later. Yet it is exactly their monolithic treatments which yield useful concepts to start with. Pickering, Latour and Heidegger all put forth visions about alternative approaches to modern science.

Andrew Pickering's history of the early British cybernetics scene (2010) may not seem relevant at first sight, but I hope to show that the experiences he is writing about resonate closely with hacker culture and the tinkering that is going on in the hackerspaces. Moreover, his conceptual framework derived from these experiences could solve a number of problems with the interpretation of hacker culture outlined above. What he is writing about is an alternative scientific practice flourishing in the cybernetics community. Of course, science is concerned with the establishment of truths and the development of technology, so another approach to scientific inquiry does have far ranging social consequences. His circle of cyberneticians — mainly Grey Walter, Ross Ashby, Stafford Beer — have theorised for fun and profit, constructed machines on their kitchen tables to test and showcase their theories, and navigated an institutional terrain that was alien to their transdisciplinary interests.

Cybernetics (the science of control and communication) appeared as an anomaly in the history and philosophy of science and largely remains so. Pickering points out that it has never achieved to find a stable "social basis" for itself, founding institutions capable of producing the next generation of cyberneticians. University departments, research centres and companies have scarcely remained afloat, even though they gathered their impulses from so many fields of

inquiry. The conclusion is that cybernetics did not fit into the division of labour reminiscent of modern institutions, of which the division between disciplines and the dividing line between business ventures and the academia are just two examples. The novelty of cybernetics was so striking that more often than not its potentialities were all but lost on a baffled audience.

The Musicolor machine of Gordon Pask was a case in point. Constructed while he was a medical student at Oxford, this device transformed music into a light show. However, as Pickering phrases it, it was designed to "get bored" (2010, 316), which meant that it had its internal dynamics corresponding to the building up of electricity inside its circuits, so that even if the music repeated itself, the Musicolor never. Coupled with a piano the player could conduct a sort of synaesthetic conversation with it. Plask and his companions assembled it from found parts and premiered it in the crypt of a Church in a spectacle performance. Next, they attempted a tour to present it to businessmen around the country and managed to install it in a couple of places as part of varieté shows. However, it never managed to become a commercial success. It did not work as a pure art form, and in their desperation they tried to sell a smaller version in conjunction with jukeboxes (2010, 317). Finally, Pask ended up using the same logic with greater commercial success in designs for the Self Adaptive Keyboard Trainer (SAKI), and later in even more complicated teacher- and pupil simulators.

Apart from that these and similar stories recounted by Pickering abound in strange and funny anecdotes, there are a number of factors which correspond closely to the life of the r0ket to be surveyed and analysed in the next chapters. They are based on extracurricular experimentation in a milieu close but still standing apart from the academia. They produce aesthetic and playful experiences based on the exploration of emergent phenomena. Their scientific, commercial, or artistic success is highly doubtful despite the immediate positive response of audiences and users. Last but not least, they are based on an inspiration of technological possibilities and techno-

scientific experiences rather than a well defined research project or a clearly sketched out problem domain. In sum, they pose a challenge to the modern categories that define the areas of human pursuit.

In order to account for these discrepancies, Pickering came up with the concept of cybernetic ontology. According to the cybernetic ontology the world is in constant flux, which makes it fundamentally unknowable and unpredictable, so the best way to deal with it is adaptation. This is in stark contrast with modern science, which starts from the premise that the laws of nature do not change, therefore the world is fundamentally knowable, and the limits of understanding can be overcome by refining models. A crucial difference between the two ontologies is that the modern approach seeks to come up with a correct representation of the world, while the cybernetic approach loathes such detour through knowledge, aiming for a correct performance instead. Pickering argues that this duality between representation and performance is the key difference between the practices that stand on the ground of these ontologies. Moreover, according to him this contrast can best be grasped at looking at performative objects that exhibit the cybernetic ontology as an "ontological theatre", making it more concrete and graspable. Since people who act on the basis of a cybernetic ontology are prone to designing machines as part of developing their theories, such machines should not be hard to find. This is why I choose to focus on a particular device that emerged from the hacker scene: the r0ket badge.

At this point it is worth to take note of the foremost theoretician of the modern-nonmodern divide in the social sciences, Bruno Latour. While his claim that a monolithic modernism has ruled over our understanding of ourselves seems to me far fetched, his writings nevertheless carry great significance in thematising the issue of modernity in a critical way. A simple compilation of the concepts developed to understand the contemporary — postmodernity, late modernity, high modernity, or even liquid modernity — draws the contours of a dreadful conclusion: that each of these formulations introduce the end of a story, namely the demise of modernity. In order to take

account of these results in a forward looking way we have to engage critically with the concept of modernity and search alternatives. We have to question dichotomies of man and machine, nature and culture, subject and object, science and situated knowledge, and so on and so forth. However, even that is only possible through a self-critical consideration of our history, which indeed is richer than the mainstream modernism that Latour targets so ardently. As Pickering have put it in his review of Latour's manifesto "We Have Never Been Modern":

> "Especially since World War II, sciences like operations research, cybernetics, and systems modelling have explicitly thematized the human/nonhuman hybridity of culture and practice — but far from inhibiting change, they have served to continue and possibly accelerate it. In this respect, then, I think that Latour is explaining too much with his conception of modernity." (1993)

My hypothesis is that hacking, more than any other practice pointed out by Pickering, is aligned with such a research direction. Authors like Söderberg and Adrian Wilding (2010) have argued that theoreticians like Hegel (for both), Marx (for Söderberg), or Schelling (for Wilding) already provide the tools for the critique of modernism Latour advocates. Indeed, there is a case to be made for a thread of philosophical thought stretching from Heraclitus through Spinoza, a version of Hegel, Bergson and even Whitehead to Deleuze — with perhaps a special spot for German Romanticists like Schiller, the Schlegel brothers and Schelling — which undermines modern assumptions in a productive manner. However, as Wilding states very precisely, the critique of modernism cannot be a purely theoretical undertaking corresponding merely to a shift in analytical stance. We live in a ("late") modern world which has to be transformed through a nonmodern practice. This is why a close look at hacking as a practice rather than a theory can help this effort. In fact despite some of his claims to the contrary Latour is also doing this through his recuperation of Whitehead, Tarde and others, and also in his work of veritable critical philosophy in which he identifies the foundational concepts of modernity and feels out their theoretical and practical limitations.

What is most interesting in his avant-garde treatise *We Have Never Been Modern* is the political critique of modernity. He argues rather convincingly that the separation of scientific truth and public opinion which is a staple of modernity is an ideological operation which guards the rational, enlightened elite from the unwashed, irrational masses. The separation of subject and object, which divides necessary, inevitable, and imperative facts from subjective opinions serves a similar purpose. The reader may be reminded here of Söderberg's argument that the political significance of hacking can be grasped in the fact that capital and the corporations have lost power over research and development. However, Latour's argument gives a deeper philosophical basis to that argument. Finally, the subject and object dualism penetrates the human being itself, disengaging the body and its affects from the "mind in a vat", as Latour calls it. There is indeed a fascination in the hacker community and in the cyberpunk cultural movement in general with the very perversity of this operation but when science becomes a source of not only fascination but also enjoyment for hackers, this is overturned. That is when cyborgs on the one hand (idealised minds in a vat) and cyborgs on the other hand (human brains with artificial bodies) take the stage. In the larger scheme of things science is the domain of nature. Scientists specialise in establishing facts through looking at nature. They guard the facts against other people in society. When the separation of facts and the social is translated onto another plain, it becomes the separation of the artificial and the human, where the science fiction figures evoked above surface again. Here Pickering's cybernetic ontology which places man, machine and nature on the same ontological plane makes sense. For Pickering's cyberneticians these are only systems of various degrees of complexity.

Although Heidegger had a one-sided interpretation where he saw cybernetics as the epitome of modern science — which I will not refute here for the sake of brevity, he is still useful for this inquiry in three regards. Firstly, he is the primary master of ontological (and ontic) analysis in twentieth century Western thought, and therefore his guidance is of pivotal importance even if

ontology is not used in a straightforward philosophical sense here, only as a metaphor for a certain type of socio-cultural disposition. Secondly, as we will see his take on technology is quite relevant in the analysis of hackers. Thirdly, as both Latour and Pickering, I will also use the dual concepts of enframing and revealing in my argument, which are better clarified at this point. Despite Latour's critique where he argues that Heidegger sees technology as pure enframing (1993:65-67), I argue that the latter's analysis indeed gives credit to the liberatory potential of technology in a way that is useful for understanding hacking if combined with the ideas drawn from hacker studies presented in the first part of this chapter.

His essay *The Question Concerning Technology* (previously published under the more inspiring title *The Framework*) departs from the consideration of the etymology of technology, which comes from Ancient Greek *techné*. *Techné* was used in a much more wider sense as "the name not only for the activities and skills of the craftsman but also for the arts of the mind and the fine arts". Furthermore, it was closely coupled to *poiesis*, the aesthetic skill of bringing forth that is essential for the pursuit of truth, which the Greeks called *aletheia* — unveiling. As we have seen earlier, both Söderberg and Himanen emphasises the aesthetic dimension of hacking and how hackers conceive of their work in aesthetic terms. Therefore, in a way the ancient concept of *techné* corresponds much more closely to hacking than the English word technology. It is worth to keep this in mind throughout the remaining part of the paper, although I will continue to use technology for the sake of clarity and consistency.

Technology as an interface between human beings and their surroundings (nature) plays a significant part in their relationship to truth. Thus, modern technology for Heidegger holds a great danger — exactly, "the danger in the highest sense" —, but also saving power. In case humans merely instrumentalise technology in the manner of enframing, they stop to "listen, but not obey", which is the necessary stance for the poetic emergence of the truth of Being. In this case technology becomes a means of concealing rather than an opportunity for revealing, and in

turn it will impose its own false truth on humans, with catastrophic consequences. In short: enframing blocks poiesis. The saving power of modern technology lies in the possibility of "our catching sight of the essential unfolding in technology, instead of merely gaping at the technological. So long as we represent technology as an instrument, we remain transfixed in the will to master it. We press on past the essence of technology." The poetic playfulness of hackers that pursues the "beauty of the baud" — as The Mentor puts it in *The Conscience of a Hacker* (1986) —, instead of mere utility, addresses the question concerning technology in a way that Heidegger calls for.

Descriptions of the case

In this chapter I describe the object of the research, i.e. the r0ket device. The presentation is spread across the following sections. "The object" section answers the question why the r0ket looks like a rocket, and provides some background on electronic tinkering in hacker culture. "Production" details the circumstances and process of r0ket design and manufacture. "Distribution" tells how the r0ket was launched to the world and to the hacker scene. These diachronic sections are followed by synchronic sections explaining r0ket software and hardware. Finally, "Use cases" describes how hackers made use of the r0ket. These chapters form the main body of material that will provide the background for the analysis.

The object

> r0ket
> The r0ket is the badge for the Chaos Communication Camp 2011 and the 28th Chaos Communication Congress.
> Besides being a shiny electronic name tag, the r0ket is an easy to use full featured microcontroller development board.

A rocket shaped electronic gadget, the r0ket device (pictured on the first page) is a double sided printed circuit board with numerous small mounted electrical components visible on the surface, as well as many holes along the sides and a small display in the middle. What clearly sets it apart from commercial products is the fact that it has no case so that it reveals all its architecture to the naked eye. This is significant in itself since it invites exploration and hacking.

The shape needs further explication. One of the prominent scenes at the major events organised by the Chaos Computer Club in recent years has been a rocket shaped statue around 4 meters high that became a sort of second logo for the organisation. For example it occupied the Alexanderplatz in Berlin during the Chaos Communication Congresses that were organised in recent years. The word "space" has also been used in naming the next generation of community workshops run by hackers: the hackerspaces. The Hacker Space Programme was launched during the 2011 Chaos Communication Camp, where the r0ket premiered as the official conference badge. The ultimate goal of the programme is to send a hacker to the moon in 23 years (23 being a magic number featuring in the *Illuminatus!* trilogy by Robert Anton Wilson and Robert Shea). The first major phase consists of placing 3 satellites in orbit around the Earth to counter the censorship efforts of governments. Of course the phrase "rocket science" is used in English in a derogative way to mean that something is too complex and confusing. In contrast, hackers aim at an elegance or at least effectiveness which provides simple solutions to complex problems. In line with this ideal the hacker organisations like the Club aim to bring serious science in reach of laymen. Therefore, the rocket is a fitting shape for the r0ket badge to echo the imagery and aspirations of the hacker community.

While the cyberpunk era of the 1990s was dominated by the exploration of the possibilities of the virtual worlds often called the cyberspace, hackers gradually turned back towards the physical during the first decade of the twenty-first century. Microelectronics are a case in point since they played a key role in kickstarting hackerspaces, as evidenced by the popularity of basic electronic

classes and programmable microcontroller workshops in the programme of young hackerspaces. Physical computing was laid out by Igoe and O'Sullivan in *Physical Computing: Sensing and Controlling the Physical World with Computers* (2004), and had a great impact on the whole computing scene.

This new framework of human-machine interaction stressed the way people behave in everyday situations using their whole body, and opened the way for exploratory research through the construction of intelligent appliances. The next year O'Reilly Media started to publish *Make Magazine* which focuses on do-it-yourself technology, including tutorials, recipes, and commentary from a wide range of authors including some celebrities of the hacker subculture. "The first magazine devoted to digital projects, hardware hacks, and DIY inspiration. Kite aerial photography, video cam stabiliser, magnetic stripe card reader, and much more." ("DIY Projects, Inspiration, How-tos, Hacks, Mods & More @ Makezine.com - Tweak Technology to Your Will", *Make Magazine*) In Europe, Massimo Banzi and others started to work on the invention of Arduino, a programmable microcontroller board with an easy-to-use software interface. This amateur-friendly microcontroller system became the staple of hackerspaces and artists' workshops and initiated a whole new generation into rapid prototyping and electronics work. To put it together, physical computing provided a theoretical area to be explored, and the Arduino became its killer application, while *Make Magazine* and similar media facilitated the spread of research results. Finally, the hackerspaces became the site for the enacting of these changes. An interesting but tentative hypothesis for future research would be to see if the whole process fitted into the bigger picture of a gradual paradigm shift which marked a move away from the linguistic turn where aesthetics served as a general interpretative tool in any discipline to a more pragmatic one founded on architecture and the body. In the following pages I gradually present a set of clues about how the r0ket can be seen to feature in such an overall process, a subtle shift of emphasis from representation to practice.

Production

The r0ket emerged from a group of hackers more or less affiliated with a hackerspace in Münich, Germany — called MuCCC. The r0ket team comprises around half a dozen core members and around a dozen dedicated members. They claim to rely on the help of more than a 100 participants when more massive manpower is needed. Some team members encountered the idea of hacker badge when they visited hacker gatherings in the United Stated like DEFCON and HOPE. They liked the idea of a badge which actually does more than just displays a name, and which is interactive in an open-ended way — what they call "hackable". It was a good way to introduce the vast number of programmers who attend such events to the field of electronics. On the other hand, they were not satisfied with these badges because people did not find them very useful after the conference. However, these badges were already advanced enough to foreshadow the possibility of developing badges that are suitable for supporting the electronic tinkering that usually goes on in hackerspaces.

That is how they were inspired to develop one with better specifications, bigger display, and more potential. After a few prototypes a somewhat similar but less sophisticated version of the r0ket was put together for the EasterHack conference, a smaller European hacker gathering that mainly caters to German speaking hackers. It was all hand soldered. This Ur-r0ket was distributed in 300 copies during that event and a 100 copies of it sold within minutes. Therefore it was proven that the developers are able to design and manufacture a serious badge and the audience is extremely responsive to the idea. The conclusion was that it had to be pushed to the highest level of the Chaos Computer Club hierarchy, which is the massive Camp and Congress events (see next section).

The principal architects of r0ket are mostly male computer science students, some of them already working in IT for companies, often as system administrators. The roughly 10% gender balance reflected the membership of the Chaos Computer Club (Blanc & Noor 2011) — which is more a software and security group — and the gender balance of the Chaos Communication Camp (Braybrooke 2011) — which is more concerned with hardware hacking. This ratio is evidently pretty low. As one of the most famous Do It Yourself hardware hackers, Limor Fried — who is reportedly a role model for many young women in engineering — has phrased it, the low representation of women in the scene is a Catch22 problem: in order to get more women into the field, there needs to be more women in the field (June 2011). On the bright side, Braybrooke notes that the sexist attitudes in the hacker movement have been slowly but noticeably changing for the better during recent years, which coincides with the anecdotal evidence from my own experience. Nevertheless, while women-only, self-organised initiatives (for example Debian Women and Linux Chix) have been set up for tackling the problem, such logic puts more stress on women in addition to the usual structural oppression that is ubiquitous in contemporary capitalist society, because they spend extra hours organising women.

The hardware and software design was handled by separate individuals, with one person taking the lead and many others joining the process with development, prototyping, testing and reporting of problems. It is clear from the records of github.com — the technical platform which was used to coordinate the software development process ("Graphs of r0ket/r0ket", *Github.com*) — that two persons (Sec42 and Schneider42) did most of the firmware coding, while many others contributed small enhancements and bug fixes when they needed a new feature or something that they wanted to use did not work correctly. A chat room interview with Sec confirmed that the coordination of software and hardware development happened mainly through loosely planned, informal, physical meetings at the hackerspace, involving only a few

people. However, there was also a closed development mailing list for virtual coordination which played a support role to the physical encounters. Prototypes were put together from common parts and customly ordered pieces on the premises of the MuCCC.

Organising the acquisition of the parts, the production process and the seed capital were as formidable tasks as the design and development of the hardware and software components. The most important factor on this side of the project was that the hackers had knowledge and connections but very limited economic capital of their own. Therefore they enrolled the help of the prestigious nation-wide hacker organisation, the Chaos Computer Club. This was not difficult since MuCCC is formally part of the CCC and site of its local branch. Furthermore, several individuals of the Münich group are instrumental to the organisation of the big hacker events and therefore hold a strong position in the organisation. The CCC provided seed funding and official adoption of the project. On the receiving end, this could be leveraged to attract sponsorship and good prices from the technology industry. r0ket team members repeatedly emphasised that sponsorship was a big part of the project which made it feasible in the first place. Even though they use the word sponsorship, the kind of sponsorship they are talking about basically means that they get products and services for free without any strings attached. In particular in the case of what is called sponsorship in the commercial world the sponsor gets exclusive rights or more often extensive media exposure and the end product becomes associated with the sponsoring brand. As described below, the CPU manufacturer gets a marketing advantage through the documentation, on the other hand, the PCB (printed circuit board) manufacturer remains anonymous throughout the history of the r0ket.

The most common and simple parts like resistors and transistors were acquired from local vendors in bulk. The display for the r0ket was chosen to be a very widely available mobile phone replacement part, the Nokia 1110. Compatible clones of this are widely and cheaply available from the Chinese market. A small factory in Münich stopped its entire normal operations for a

day in order to print the circuit boards of the first generation r0ket and charged only the price of the production cost. According to one of the videos documenting the factory operation mistakes were made in the design which had to be corrected by "thousands of helping hands" at MuCCC. The CPU is from NXP Semiconductors which gave everything for free. The explanation for this from the r0ket team is that they have just put out this processor on the market and they need a "killer product" which showcases its capabilities and creates a user base of people who know and like to program it. Since they gave the CPUs for the second generation r0ket as well, it seems that they are happy with the deal.

The USB connector and some other parts needed to be soldered on the r0ket after it came out of the factory. This was a major operation that r0ket people remember vividly and cite often when they discuss their project. They called on volunteers from the community around the MuCCC through the mailing lists and Internet Relay Chat channels. Although they were amazed that so many people attended their call — one person mentioned that individuals were travelling to Münich from other cities only to solder r0kets — even the participation of around a 100 people was not enough and core members had to spend nights and days soldering before the deadline to finish the procedure. One member proudly stated that "I probably held most r0ket devices in my hand during that time". Not only the people but also the space was overloaded with r0ket boards, to the extent that it was hard to find a horizontal or vertical surface which was not covered in r0kets. The mission obviously stretched the capabilities of the hackers.

The free printing facility was not available for the second round of r0kets, however. The second challenge faced by the r0ket developers was that the volunteers who worked on the first batch were quite tired of assembling many r0ket parts by hand. In order to still keep the price below €30 and to meet these two challenges, they asked help from one of the most prominent figures of the hackerspaces movement: Mitch Altman. He is credited with teaching numerous people how to solder and his teachings are the basis of the ubiquitous "Soldering is Easy" by Andie Nordgren

(Nordgren 2010). He is also the founder of one of the biggest hackerspaces, the Noisebridge in San Francisco. Mitch suggested the r0ket team to commission the Chinese manufacturer Etonnet to print the r0ket, purchasing most parts and assembling most parts on the PCB. r0ket team members cited three reasons for the decision to contract Etonnet for their project: (1) although they trusted Mitch they did also not have much time left to search and check other manufacturers, (2) in contrast with the perception of the Chinese labour market, Etonnet seemed to have good working conditions and fare wages for workers, (3) the prices and the quality of the samples Etonnet could give them were beyond their expectations (lilafisch & Stefan 'Sec' Zehl, 2011). In fact they were amazed that the manufacturer even tested their design against extremely high and low temperatures and provided them with a sheet showing the results of the test including what percentage of r0kets are expected to malfunction under what conditions.

In terms of virtual communications infrastructure, technical interrogation shows that the r0ket team started to prepare its public coordination infrastructure roughly a month before the Chaos Communication Camp where the first generation badges debuted. A few minutes before midnight on June 13th r0ket.de was redirected to a new wiki site (according to Whois info from DENIC, 2010), and the next day saw the first editing of the front page (according to index page history). The same day saw the first posts on the r0ket account on soup.io which was the prime promotion surface through which developers kept up the expectations of the audience, and later advertised the cool things that people were doing with the device.

The soup is a tumblr clone that has been developed in the Metalab hackerspace in Vienna and now operated by a spin-off company (euphoria GmbH). Soup is less than a blog, but more than a profile, providing an easy mechanism for gathering texts, images, videos and twitter updates on a single page, and also reposting them between different soups and other social media. In short, it is

a micro blogging platform. r0ket also has its own mailing list and chat room (IRC channel) hosted on the MuCCC servers. The firmware of the r0ket is on github.com (registered a few days after the others, June 18th.

Distribution

As was hinted above, the two generations of the r0ket device have been distributed in different ways. The first one was the official conference badge of the 2011 Chaos Communication Camp (August 10-14), of the two major camps in the European hacker agenda. This gathering attracts around 5000 hackers each year. In recent years it takes place on the grounds of the Finowfurt airport, including landing grounds and an open air aeroplane museum. Thus some hackers arrived on an aeroplane and touched down a few meters away from the tents where people lived. There were a variety of different tickets available: the standard ticket cost €140, the business tickets €850 and there was a possibility to write to the "friends team" if somebody wanted to get in for free for a good reason. The 2011 camp prices included €15 that went to cover the cost of the r0ket badges, in addition to some extra money from the savings of the Computer Club. Haying said that, the security of the camp was relaxed, so some hackers simply walked through the forest to get to the campsite.

Since the second generation r0kets were distributed at the Chaos Communication Congress it is worth contrasting the two events, also because the distinction does bear relevance to the different distribution methods. "Hacking for fun and profit" is a popular hacker slogan and it is possible to argue that the Camp is more for fun and the Congress for profit. That is to say, the Camp concentrates on creating a friendly community environment while involving newcomers in the scene, while the Congress is a bit more business-like and has a stronger technical/academic element. The main action during the camp takes place in self-organised "villages" of tents, while

at the Congress the focus is on the lectures, talks and presentations. That being said the Camp also includes a full conference programme spread across 3 giant hangars, while the Congress has a hack centre where groups can occupy tables and chair and rooms and they can organise their own activities. In line with these differences the Camp is organised in a nonprofit way, while the Club seeks to make a profit from the income on the Congress.

Visitors with a valid ticket simply got a r0ket at the entrance in small plastic cases with the printed Camp brochure and the wrist band. The case included various components as well as assembly instructions (see Fig. 1. & Fig. 2.). Basically, users had to connect and stick the battery and the display to the r0ket body. Once done that they were instructed to set the name that they want the badge to show on the display — after all this is the primary purpose of a conference badge — and the privacy setting for the wireless mesh network that the r0kets create amongst themselves. The latter includes the choice of turning off the radio, using it anonymously, or sending the configured pseudonym along with the signal as well. After doing all that they were welcome to use, hack, and keep the r0ket. The default software and hardware capabilities of the r0ket will be described in subsequent sections. The last notable fact is that the first generation r0kets were not and are still not available outside of the single event where they premiered, which means that owning one is a good sign of having participated there. Of course, as I will describe in more detail later, since the r0ket is an open source product, it is relatively easy (according to team r0ket: "no problem") to make a similar one.

Exactly because a conference badge is always unique for the conference (see Fig. 3.), the r0ket could not be the official badge of the Congress. However, the r0ket is more than a conference badge — as mentioned earlier, it is also a prototyping solution. The r0ket team started to negotiate with the Computer Club about the idea. They wanted the Club to pre-finance the production and sell the r0kets for the price of production at the Congress from a stall. The difficulty was that the Club was doubtful about whether hackers at the Congress would buy all of

the r0kets. The parties reached a compromise with the help of Pollin, an online store which sells electronic parts on the Internet for technology enthusiasts. The Club would pre-finance the production of a thousand r0kets (seed capital!), and 700 would be sold at the Congress, the 300 others going straight to Pollin after the Congress. The safeguard was that Pollin would buy the left over devices from the Congress sale in case there was any. This scheme ensured that all the r0kets would be sold and the Club would get back its money that it invested.

The conference happened from 27th to the 30th of December (2011) at the usual venue, the Berlin Congress Centre at Alexanderplatz, drawing around 4000 visitors. The r0kets were sold during the zeroeth day when people come to pick up their tickets and inhabit the space, and also during the first day of the conference. The price was set for 30 € which according to the r0ket team matched the production price almost to a penny. Once again the r0ket team was under enormous pressure because once the stall opened there was instantly a big line in front of it and stocks sold out instantly (well, it took about an hour to serve everybody). They were also selling Flame extensions with the r0ket for €5 These can be snapped on the r0ket, comprised by a flashing led and a piece of plastic reflecting the light, with hacker-related emblems milled on it. The rest of the conference was saturated by various hacks, games and lectures about the r0ket, detailed in subsequent sections. On April 18th, 2012, the r0ket team announced that r0kets are on sale at Pollin for €39,95 ("Modul rOket"). According to German speakers commenting on the chat channel of the project the description was not very good and it missed the point of r0ket being a development board, but they were happy with the 20 page German user manual that the company produced for the device. As of April 27th, 2012, the item is still on sale.

Hardware

Just like the software (and firmware) that runs on the r0ket, the hardware design is also open source. In practice the latter means three things: (1) that the schematics of the printed circuit board (PCB) that shows how to connect the components, and (2) the specifications of the components themselves are documented on the r0ket website (r0ket wiki contributors, 2011). Additionally, (3) according to answers to my query on the chat channel, it is licenced under an open source BSD licence, although I found no conclusive evidence of this. As stated above the r0ket is more or less a small computer. The hardware parts of the r0ket are easily available on the market. The display is the ubiquitous Nokia 1200 that is used in many old mobile phones. Compatible clones manufactured by many businesses. Thus the choice was motivated, amongst other factors, by its easy availability and good documentation. It is a green monochrome LCD with 96×68 resolution. The connector is the same as the Nokia 1600, which is a colour display. The default firmware for second generation r0kets supports the colour display out of the box even though the r0ket is shipped with the same monochrome LCD.

There are various ways to communicate with the r0ket. The main input device is a 5-way joystick (left, right, up, down, up and click) below the display. This is used to navigate the configuration menus of the device. There is also a light sensor that switches the display to negative in brighter environments so it is easier to see it. Like all sensors, this can be used as an input device. The most interesting input is the radio frequency receiver-transmitter on the board, which is basically a small radio. When r0kets are close to each other — according to tests conducted in H.A.C.K., the distance can be around 10 meters — they pick up the signal of each other and are able to send signals between themselves. When a lot of r0kets flock together, they create a so-called mesh network that can cover large areas. The gatherings where r0kets were distributed were good playgrounds for this. For instance the high scores of the Space Invader game included with the

r0ket (see below) were synchronised across the whole airport during the Camp. Moreover, the r0ket signals can be picked up by commercial RF receivers from afar and processed by real computers if needed. The technology is similar to the well-known WiFi (802.11 family) that is used in laptops and mobile phones — for example it operates on the same frequency (2.4Ghz), but it is much more primitive. The communication protocol is based on a community-developed standard called OpenBeacon (see openbeacon.org). Finally, there is a micro USB (2.0) connector used for charging the battery and uploading to the dataflash storage (which around half megabyte capacity), or flashing the device. This can also be used for communicating with the r0ket in real time.

The computations are done by a 32 bit ARM microcontroller, the Cortex-M3. The ARM architecture is very popular at the moment with embedded devices. The computer manufacturer Asus and others have recently made a laptops using this technology. As mentioned above this CPU is a new product that the manufacturer wants to spread on the market. For this reason it is strategically important to have a prototyping board with which engineers can experiment with, and the r0ket aspires to fill this gap in the market.

What makes the r0ket a development board, however, is that connectors that can be used to control the device from the outside, or to control other appliances hooked up to the r0ket. These are basically small holes which accept pins or wires through which electricity can pass. Most users connect simple sensors (touch, humidity, distance, etc.) for input and LED lights for output, but as described in the “Use cases” section others can connect virtually anything. Even washing machines, helicopters or routers can be coupled with the r0ket through these connectors. The result is an assemblage through which the r0ket becomes an integral part of another device.

Here it is worth to take a small detour and contrast the r0ket with its main competitor in the hackerspaces scene: the Arduino — another completely open source microcontroller development board. This author worked with the Arduino on some projects, and interviewed inventor Massimo Banzi in his workshop space in Milan, Italy in 2008. Arduinos are very popular in hackerspaces, and it is thought that there are more than 300.000 of them in use, with countless others manufactured by enthusiasts or cloned and sold by companies. This is perfectly legal since — as stated above — the hardware design and the software is free and open source just like in the case of the r0ket. Massimo said that he thought of making the design available for everybody as a contribution to the community from which he gained so much fun, knowledge and eventually his living. It was designed in 2005 and manufactured by Smart Projects, Massimo's company. The company makes art installations and industrial projects often utilising the Arduino, and Massimo makes many workshops teaching rapid prototyping with the device.

There are three principal differences between the Arduino and the r0ket. Firstly, the Arduino was designed in the context of a start up company that had a comprehensive business plan built on generating profit from the design, which worked out very well indeed. In contrast, the r0ket was never conceptualised as a product and the r0ket team had no plans to commercialise on its success. Indeed, in my interview I asked them about the possibility of making a product along the lines of the Arduino out of the r0ket, and the team members in the interview said that they think it could be viable but they have no ambitions to pursue it. In fact after all the energy they have put into the project they were happy not to work on it for some time after the Congress. Secondly, the r0ket leverages the advances in technology that were made in the meantime and boasts a much faster Central Processing Unit (CPU) than the Atmel AVR in the Arduino. This was mentioned as its chief advantage over of r0ket by hardware specialists in the Budapest hackerspace (interview with dnet and mrtee). Of course the greater processing capacity makes r0ket suitable for a greater number of projects, and the fact that the price is almost the same is

quite important in this regard. For instance the supplier Pollin Electronics mentioned above, which now sells r0ket for €39,5 sells the standard Arduino UNO for €27,95 ("Arduino UNO SMB Edition"). Third, the Arduino comes with more supporting documentation and services and it is generally targeted at "artists" — often a derogatory term in the hackerspace scene for people unwilling to learn — for making interactive artworks.

Software

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) simply turns the device into USB storage mode, used for data transfer.

The more interesting functionality comes with the programmes included on board called l0dables. There are seven of these pre-loaded on the r0ket. I claim that they gather the most important sources of hackers culture because they were designed to make hackers enthusiastic about the r0ket and its possibilities. 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 if the r0ket is working properly.

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. Gaming is a typical hacker pastime — for example the inventor of the sci-fi genre popular with hackers, William Gibson, thought of the word cyberspace while looking at teenagers playing such classic games in an arcade hall. The 5-way joystick on the r0ket is ideal for playing this game.

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 an 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.

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 them 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, 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 outlines by these application: there was games and mathematics, and now (grey hat) security research.

ROCKETS is similar to the "messages" functionality presented above, 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 group — it can easily happen during a hacker gathering that you are only a few meters away from somebody that you know online but do not recognise based on their face. 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 your good friend or idol.

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 authorities, the corporation or their other enemies capturing their messages and listening in on the conversation. Therefore they use the rather popular Pretty Good Privacy technology 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 though 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.

Use cases

This section highlights a selection of hacks undertaken with the help of the r0ket.

Making custom cases for the r0ket has been a very popular project during the Chaos Communication Camp, perhaps also because of the outdoor setting and the laid back environment. Semi-regular knitting workshops were held where people could knit their own r0ket cases, as in Figure 4. Knitted scarfs were also prepared for the old aircrafts stationed at Finowfurt. Other hackers used 3D printers which can print objects layer by layer from plastic following a digital model fed to them by a computer. Figure 5. shows one specimen. 3D printers — a relatively new obsession — are one of the few things the visitor surely finds in any hackerspace. As Söderberg argues, hackers question the monopoly of governments and corporations on research and development. The development of 3D printers goes beyond that by questioning the monopoly on factory production, since it provides a versatile way to make a wide variety of objects from various materials. Moreover, implementing the latest trend in

contemporary capitalism epitomised by Nike's offer that each customer can design the graphics of their shoe before it ships, with a 3D printer each item can be customised at will, and not just the surface decoration but also the shape of the object.

Another common hackerspace project that become popular only in the last few years is the quadrocopter. These designs were inspired by the rise of military drones, most vehemently utilised by the US Army. After they became a usual topic of media stories on Afghanistan and other countries, hackers decided that they wanted their own ones. At first it seems these drones are high tech military grade technology, but hackers found an easy way to replicate them by using an X shaped body with four rotors. The badge was used to control some of these. This highlights the versatility of the r0ket as a rapid prototyping tool: while a quadrocopter built in such a way is not the best quadrocopter, it is much easier to assemble and construct then models which rely on different parts to solve each problem, all of which have to be coupled together physically, electronically and logically. That is why the r0ket version is called a prototype. Some hackers took the same concept further and used 3 r0kets to build a remote control car — two r0kets sat on the chassis, powering and steering the two pairs of wheels, while a third one was used as a remote control, communicating with the two on-board r0kets separately through the RF signals described in the hardware section.

Jeff Keyzer came to the Chaos Communication Camp with an almost ready open source Geiger Counter Kit, and some radioactive material. He appreciated the r0ket enough to include it in his workshop where he teaches people how to assemble the kit he sells. The counter in the kit is making beeps and flashes for every count, but it doesn't actually count the radiation beyond this. The r0ket was ideal for receiving this raw data real time from the counter and keeping track on it on the display, while storing the data on the USB memory. The idea was very successful and the workshops were repeated at the Congress ("Geiger Counter Workshops", 28C3 Public Wiki). A similarly surprising idea during the Camp was to measure the electric currents in human bodies.

The SCOPE l0dable does exactly that: it displays an oscilloscope (electricity meter) on the screen which measures the electric field on the hackerbus (the little holes to the left and right of the screen). Thus if the user puts their fingers on the right holes the scope will measure the electricity in the body.

The Chaos Communication Congress provided many novel r0ket hacks, mostly revolving around the wireless functionality. The r0ket team spent much time tweaking and programming r0kets, but they also found time to prove that they are not only expert engineers but also excel in show business. The scheduled lecture about the r0ket ended with a mass pong game involving more than a hundred participants jointly controlling the opposing pads on the projected screen. Although it was not mentioned, but this was probably a reconstruction of an experiment organised by cybernetics pioneer Loren Carpenter in 1991. In any case, it sent the audience cheering at each touch of the ball. Many different versions of the same game were born. People in the basement of the Berliner Congress Centre were playing pong against each other in teams, pairs and against computers — using the r0ket as a remote controller and a laser beam to draw the graphics on the wall, or simply using the r0ket's built-in display.

The second major achievement of the r0ket team was to organise the Hacker Jeopardy, which is said to be traditionally the most animating part of the conference (unfortunately only for German speakers though). It is the clone of the popular television show Jeopardy, where two competing participants have to answer the questions of the host by saying another question which has the same answer as the first one. Since it can be up for debate if the two questions really have the same answer, somebody has to decide on it. This is where the r0ket comes in to the picture. During the Hacker Jeopardy audience members could vote if they accepted or declines the answer gave by the contestant. However, r0ket was also present on the content side of the competition.

During each round, players can choose what kind of question they want to answer from a range of 5 topics and they can also choose the difficulty level of the question, which corresponds to the amount of points they get for the right answer. r0ket was one of these categories.

Another more sophisticated toy project was started during the Camp almost from scratch (robfitz 2011b) and finished during the Congress (robfitz 2011a), lead by Robert Fitzsimons from Part Fusion Electronics, Dublin (robert 2012). Laser Tag is a game similar to Paint Ball apart from that people are not shooting at each other with paint balls but infrared guns, and each participant has a sensor that beeps when he is hit. The mission was to build such a laser gun with an integrated sensor and beeper. There has been three or four prototypes and a workshop produced during the Chaos Camp and tests showed a range of almost 50 meters. The r0ket was used to control the infrared sender and receiver, the microphone, and counting the number of hits sent and received. A custom r0ket firmware was produced to correct some unsuitable behaviours in the original operating system. The process was helped by advices and pointers from various Camp participants including Mitch Altman (mentioned above) and the r0ket team. At the Congress Fitzsimons was already able to make a workshop for 20 participants where they assembled more copies and played the game for much fun.

During the Lightning Talks of the Chaos Communication Congress Tobias Weyand and Christian Buck presented okr0ket, a dating application for the r0ket (2011). It is based on asking a few geeky questions about the user and then broadcasting the answers in a compressed format to all other r0kets in the vicinity. In the event of a match found the red led (which is actually green on the first generation r0kets) on the lower left side of the r0ket starts blinking. Partial matches are displayed on the screen with their match score. Questions thematising hacker lore like the significance of number 23 and 42, encryption, lolcats and the "editor wars" between the two popular source code editors vi and emacs. The latter is an interesting instance. Both editors have a long ancestry going back to 1976. They are associated with legendary hackers Bill Joy (co-

founder of SUN Microsystems) and Richard M. Stallman (inventor of free software with the GPL licence). Since their design philosophy is is in diametrical opposition to each other they are ideal arch-enemies of each other. The Chaos Communication Congress even included a workshop called "NPC — Nerds' Pissing Contest" where teams contested each other in solving text editing tasks with their favourite editor (Kellermann & klobs, 2011). Ironically, the "line editor" sed came out on top (developed by Lee E. McMahon of Bell Labs fame in 1974), so once again the rivalry between emacs and vi could not be resolved. Perhaps this explains why it is important for geek lovers to be able to agree on their editor of choice.

The experiment of largest scale involving the r0ket during the Congress was a mass tracking exercise. The r0ket team placed powerful receivers around the conference area and read the nicknames broadcasted by the r0kets in real time. Using a triangulation between the receivers it was possible to track the movement of r0ket users who have chosen to take part in the experiment by setting their RF privacy setting to zero. Characteristically, the hardware was brought to the conference but the software framework for the visualisation took shape only during the event and after (see "r0ket tr4cker — 28C3 R0ket Tracking" in references). Later all the traces were published for people to play around with (Sec, 2012).

On a lighter note, the r0ket team celebrated the new year by launching a rocket with the r0ket — a fire cracker with the help of a purpose-made electronic device controlled with the r0ket ("Launching Rockets with r0kets", 2012). In the new year r0ket workshops are held in Beijing, China during the Maker Carnival, "China's First Global Mass Creation and Open-source Sharing Faire" (Maker Carnival, 2012).

Analysis

Liberalism

As with the other considerations, rather than talking about the attitude or motivations of hackers in general, the analysis will start with the concrete ethnographic details and then abstract their ideological effects.

Expressive individualism as defined by Charles Taylor (2007, 473) is an ideal of liberal freedom in which each person finds their own way of living and expresses this mainly through the use of consumer goods. While the consumption of consumer goods is a highly individualistic affair, the same individualism is reinforced on the production side, even in the sphere of knowledge production which is a highly collaborative field by its nature. In the academia — which Himanen establishes as the model of hacker culture versus the monastic order of capitalism and the Protestant Ethics — scholars have to sign their articles and they have to submit their own Curriculum Vitae to potential employers. Collective results like the reputation of universities is assembled and aggregated from these individual achievements through the instruments of publication frequencies, impact factors and postgraduate employment statistics. As Taylor notes (as early as 1992), this is rooted in the Romantic concept of the genius as an authentic author. Against this background the authorship of r0ket stands out as an example of what I would call **collective expressivity**. It is plain to see from this term that I am not trying to argue a mere anti-liberalism but a reconfiguration of liberalism through collectivist practices. However, it is noteworthy that since individualism is a core value of liberalism, the result can scarcely called a new or different liberalism.

The r0ket is officially signed by the "team r0ket" on the cover page of the r0ket wiki, the website of the device, and similar signatures appear elsewhere in various documentations and communication spaces. Although there are individuals who could be called "chief architect" of the r0ket software code or hardware design, or others could make the case for being an

"acquisition manager" and so on, no individuals make such claims over parts of the work done. This is unusual in mainstream software development, and very different from hardware development practices where products are usually tied closely to the brands maintained by corporations. Moreover, it is even against some of the ingrained habits of root hacker cultures like the demoscene, where virtuoso animations are tied to group names and also individual authors. Even large collaborative projects like the Linux kernel or the Python programming, or Wikipedia for that matter — which is not even a technical project per se — have a "face", sometimes called a "benevolent dictator" like Jimmy Wales of Wikipedia, although articles are credited "Contributors of Wikipedia" of course. One small concrete example of this authorial dynamics in the r0ket project is that the presentation of the device at the Chaos Communication Congress was officially registered to be conducted by lilafisch and Stefan 'Sec' Zehl. However, lilafisch could not be there and she was substituted by a guy from the r0ket team. This caused no major disruption since many individuals had adequate knowledge and skills to make the presentation. However, even though there is a possibility to change the documentation after the event took place, nobody bothered to take the necessary steps. The person of the presenter has not been perceived as relevant or import factor — which differs greatly from the perception of authorship amongst most artists, for example.

Collective authorship is not a unified concept either. Collective authors can take a number of forms, the most common being a long term collective which builds a reputation over a greater span of time through the cumulative effect of authored work which form an oeuvre. Again, demoscene animations are a good example which are signed by collectives which compete against each other inside the scene from year to year, and which have a more or less stable membership. The fact that the r0ket team, or team r0ket, is named after the product eludes this logic and directs the attention back at their creation: the r0ket device. The tactical advantage of choosing such an authorship is that it enables enrolling many people, since the light-weight r0ket team

identity requires a low investment from contributors. The strategic disadvantage is exactly that it does not fit into the logic of expressive individualism, therefore it is hard to capitalise on it later on, or even just enjoy the experience of authentic expression which is more or less tied to a strong and well-established identity. Of course, as Johan Söderberg points out, such moves of obscuring authorship can be interpreted along the lines of de Certeau's cultural studies as subversive or resistant practices against the prevailing cult of the genius — fan fiction being one example.

However, the attribution in the documentation is inconsistent with the signage on the body of the r0ket PCB itself. There it is stated that r0ket is "Designed by CCC", with the CCC logo and a QR code (two dimensional barcode) pointing to the r0ket wiki. The address of the r0ket wiki is also written with letters at the same place. Furthermore, it is significant that while there is a r0ket.de address registered on the Internet, it redirects to r0ket.badge.events.ccc.de. It is obviously easier to advertise r0ket.de than r0ket.badge.events.ccc.de because it is a shorter address that is easier to type and memorise, yet the r0ket team decided to focus on the longer address. This setup emphasises the important of the link with the CCC. Putting these two things — the attribution in the documentation and the signage on the device itself — together, it seems that r0ket team members are implying that the r0ket team is more or less a working group of the CCC. Nonetheless, hackers can be active in the r0ket team without formal membership in the Club, just as they can attend most CCC events without being a CCC member.

As described earlier, the Chaos Computer Club was instrumental in the production, distribution and marketing of the r0ket, and indeed the r0ket was the "official conference badge" of the Chaos Communication Camp in 2011, and it visibly hosts the r0ket website under the address "r0ket.badge.events.ccc.de". Moreover, most of team r0ket belongs to the "crew" of the MuCCC hackerspace, which is the physical location and social milieu in which much of the development and assembly process took place. MuCCC is also called MuCCC because it is the official space of the Münich chapter of the Chaos Computer Club. As noted, such "sponsorship" often comes

with an attached marketing string. To sum up, no Romantic authorship was attached to the r0ket, which means that it stayed relatively open to outside contributions and free from invested identities. On a final note, the endorsement of the Chaos Computer Club traditionally means more the acknowledgement of quality than the claim of authorship, since the Club has been set up to serve the hacker community and represent it. In this sense as much as the r0ket has been endorsed, it has been endorsed as something that the hacker community can call his own.

As for the cultural context, one team member described the r0ket as a "contribution to the community", which testifies to the strong collectivist logic in the hacker scene. Individuals and groups do get judged based on their contributions, but their contributions are free for everyone to benefit from, and usually made in collaboration with others anyway. In such an environment, individual expression is measured less by its difference from other achievements — although originality is definitely valued — than by their positive impact on the community as a whole. A counterexample would be bohemian artists around the Montmarte in the 1920s, where one painter would be valued primarily for the original qualities of his products, before acknowledging perhaps that she opened a new avenue of creative expression for her peers. Similarly, idiosyncrasies of speech and clothing would be seen as signs of self-proclaimed genius and creative freedom more strongly than contributions to the collective linguistic wealth of the scene, or to bohemian fashion.

Another counterexample, now from the r0ket time-line, are the workshops lead by Robert Fitzsimons at the Camp and Congress, developing a laser tagging system around the conference badge. Fitzsimons is a real entrepreneurial subject — as defined by Styhre 2005 — who is marketing himself as a tinkerer (*Part Fusion Electronics*, "About"), and apparently draws financial benefits from his teaching work as well. In the blog posts about how the idea came to him, and how he developed, tested and championed the project, he gives credit to a lot of other people, but it is clear that he mainly undertook the development single-handedly. However, the

website is called the same as his company (Part Fusion Electronics) and the post reads "r0ket Laser Tag with Robert Fitzsimons". This is not anything special in the hacker community, but simply a manifestation of a bit different dynamics inside the scene. The design for the laser tag system is of course absolutely open source — indeed, most of the information online about the workshop is the technical description of how to make the module.

The discrepancy between the authorial practice of the r0ket team and Fitzsimons may be explained by a great number of factors, such as the differing complexity of these projects, the distinction between "base research" and its applications, the financial circumstances and the civilian vocations of the authors (most of the r0ket team being university students) and the sources of funding for the project (Fitzsimons bought the parts from his own money). However, at the moment I want to draw attention to the position of the two authors in the hacker community. At these gatherings the r0ket team members were "at home" both in the sense of being mostly in their home country, Germany, and in the sense that their space and some of them are officially part of the Club which organised the events, so they need less introduction or promotion to being recognised. Moreover, at least the core members seem to be heavily involved in the life of the hacker community, for instance as organisers of the Hacker Jeopardy game, as mentioned above. When Himanen writes that caring makes the real hacker hero, he is quite right. Central people in the hacker community are often distinguished by the amount of (often anonymous) energy they put in to other people, organisation, hardware and software development, logistics, etc. just to make the whole hacker scene work. These contradictory tendencies could be formulated in familiar terms (from Garett 1968) as the dichotomy of freeriders and contributors in a pool. However, the contradictions come together in that the whole point of the work which people in altruist roles perform is exactly to enable the other — still highly productive, supportive and generous — activities to take place. Concretely, the large scale collaboration around the r0ket happened with the view to enable other hackers to use the

r0ket for their own purposes, even for making money, but most importantly for contributing in their own way to the hacker community. The workshops of Fitzsimons which gave other people the opportunity to learn, work on electronics, and most importantly to spend their time happily — also by playing laser tagging games — during the Camp and Congress absolutely fulfil these criteria.

The r0kets were initially distributed basically as gifts to the visitors of the conference. It is true that a portion of the manufacturing price was built into the ticket prices, but as we saw earlier the research and development effort was not subsidised and half of the production value came from sponsorship. John Söderberg argues that the free software ecology in the hacker community is not a gift economy in the Maussian sense, but a true — I would say communist — gift system in the sense described by Jacques Derrida (1992). The difference is that while in a gift economy the contributors expect a gift of equal or greater value back — either in a liberal individualistic way based on personalised ties between specific people or according to the logic of general reciprocity from random individuals —, a true commons is based on excess as it advocated by Georges Bataille. It is arguable that the gifting of the r0ket worked in the same way, sometimes even on the part of the industrial manufacturers (as described in the Production section above for instance in connection with printing the circuit boards). Still, as we have seen, the other point of Söderberg is worth keeping in mind, that increasing the general intellect of the technological community is also helping freeriders who capitalise on it.

The logic of the commons in its version of this "true" gift economy is based on a positive feedback loop, set in motion by inventions like the r0ket. The initial gift will set in motion a series of future gifts, all which enrich the commons further and increase its potential productivity, inspiring even more contribution. This is why "base research" such as the r0ket is immensely

important for seeding future projects, and at the same time it has to come from "true heroes" with above-average technical and project management expertise as well as above-average "caring" for the community.

Liberalism is often criticised for not taking into consideration the structural limitations on the proposed equal rights of individuals. Free software is lauded for creating a knowledge commons of programs and instruction libraries that are free for everybody to use, along with the documentation that helps individuals to improve their skills. Himanen (2001, 81), Raymond and Söderberg (2008, 96) all raise the question whether such a commons would be operational in the "flesh reality" as well. As I wrote above, looking at the r0ket is a step towards answering that question, albeit staying within the confines of technical research and development. Firstly, hacker gatherings and especially hackerspaces like the MuCCC add an embodied community to the knowledge commons mix. Concretely, if an individual would like to engage in programming the r0ket (which is mostly done in C, the most popular low level programming language), she can visit hackerspaces for free where she can get tutoring and find a supportive community. For many people this is very important for learning, that is, to actually live with the opportunities offered by free software and the bundled documentation. Not everybody can learn only from books and online forums.

Secondly, the r0ket is not merely software but it is an assemblage comprising of hardware and software components (among other even more immaterial things). Thus in order to build a r0ket, or build something using the r0ket, one needs more than knowledge and a computer. Electric components, instruments like the multimeter for measuring voltage, currents and resistance, soldering irons of various precisions, etc. are all necessary for hacking hardware — or doing rapid prototyping, to use the industrial terminology. The hardware workshop which can be found in most hackerspaces includes these tools and more. Thus, hackerspaces like MuCCC complement the online availability of software code and documentation with an embodied peer community

and a workshop space. In case an individual with enough time on her hands would like to rebuild the r0ket or use it as a part in a more complex construction, she can find a hackerspace in almost any major city (at least in Europe and North America) where she is furnished with the concrete material and immaterial necessities for the pursuit. Perhaps the best way to put it is that immaterial knowledge is complemented with material infrastructure.

In the final analysis, hackers do seek a kind of personal, individual, self-expressive satisfaction which can be theorised in liberal terms as an experience of individual freedom. However, their personal satisfaction often feeds on their ability to work together and nourish a commons, which leads itself to a more communistic interpretation. Furthermore, through teaching each other and providing the physical and logical tools for learning, they go beyond a formal liberalism which defines freedom as equal opportunity in the abstract. Therefore, they combine the focus on freedom and individual rights from liberalism with the leftist affinity for voluntary cooperation and some form of social justice. These could be analysed in a more nuanced way, taking into account the political tendencies of libertarianism, anarchy and others, but let it suffice to say here that they adhere to some liberal tenets while going beyond the limitation of the framework as well.

Capitalism

There are many ways in which r0ket and its relation to capitalism could be analysed. In order to incorporate widely divergent perspectives I focus on two aspects in this section. One is the "hard" cash flow around the r0ket — in line with the programme of looking at the networks in which the device is embedded — and the other is the "soft" construction of subjectivities.

Looking at the cash flow around the r0ket, it is evident that companies reaped profits from its construction. Schematically, hackers payed for the r0ket — through the ticket prices at the Camp and directly for the product itself at the Congress, finally ordering it online — and their money went to a cheap Chinese manufacturer (Etonnet) and a Polish online reseller (Pollin). It is not an exaggeration to state that the German hackers tinkering in their lab were able to tap into and exploit the most up-to-date globalised circuits of capitalist production and distribution when they needed so, getting very flexible offers and good deals. The key to this was not their savviness as seasoned businessmen but their access to a community which included at least a few experts in organising small-scale factory production. While it might sound trivial, it is worth to take the ability to get good deals into account since this kind of knowledge is essential for the operation of firms who are in the same business, a key to their competitiveness, and because of this such information is relatively well guarded and hard to find.

The evolution of the r0ket can be seen as a series of steps from the hacker underground to the mainstream. Began as an experiment in the relatively low exposure (but still public) and small scale EasterHack, it premiered at the Chaos Communication Camp — a big event in the hacker community — as a conference badge. Later in the year the second generation of r0kets were sold more like an ARM development board for rapid prototyping at the Chaos Communication Congress, the more serious event of the two CCCs which also gets more exposure. Here attendees already had the opportunity to buy it and spread it to people who did not attend. Finally, 300 items went to the online store with which r0ket officially integrated into the market. While marketed, the r0ket did not loose all the air of a collector's item or a small scale artwork, since no more copies were made and at the moment there is no organised continuous production or plans to release upgrades of the product line as a third generation r0ket. Another nuance is that Pollin caters for consumers more or less loosely connected to the hacker scene or at least the hackerspaces, so the exposure is actually not that great. r0ket team members have not interpreted

this process as commercialisation or the cooptation of their original ideas. Rather, they framed it as a successful process which led to the greater availability of the item. Such democratisation allowed people who did not get to the aforementioned events — for example because they live far, would not afford ticket, had other commitments, etc. — to acquire their own r0kets.

In the final analysis I think that there is a good potential in the r0ket for medium term capital accumulation in case a company decides to grab the design and start to manufacture it for the consumer market. During the course of its short life so far, it became the perfect post-Fordist product from a number of perspectives. It is aimed at a niche market — which is almost the default in post-modern marketing. It has been introduced to that market through a clever marketing campaign which gave free copies to well-connected participants of the given scene. It is endorsed by the Club which is a key player and opinion leader in the hacker world. It is produced by people whose motivations are altruistic, thus lending authenticity to the product. Its first two generations are held highly as rare items. It is composed of cheap parts which have to be assembled in a quite complex way. Its main value is in the knowledge that went to it. It is not merely a technical design but it is also immersed in a cultural framing through its shape, default software, and history which enhance its potential market appeal.

Yet the same story can be told backwards, contradicting these earlier conclusions. To start with the most outstanding point, the idea of the Chinese manufacturer conjures up images of hyper-exploited mass workers at companies like Foxconn toiling in the service of immensely profitable Western brands like Apple (Bonnington 2012). However, in this case Etonnet — the manufacturer of choice for the r0ket — is a reputable small-scale employer and r0ket team members are travelling to China this year to visit the premises and get to know the workers there. As for the cash flow side of the story, I have described earlier how the price could be kept below the market value of similar devices by enrolling sponsorship from various sides. The mind of the r0ket, the microcontroller was given for free by the manufacturer to support the making of both

r0ket generations. The PCB of first generation badges was printed for the cost of production by a company. In fact if we presume that hackers actually needed or at least wanted to have such a gadget, and it was not merely a clever marketing trick to get money out of their pockets, it could be argued that the whole r0ket project was a clever scheme spun by the hacker community to get a lot of the parts for free from capitalist companies.

Finally, there is the core issue of evaluating the volunteer labour which is undoubtedly the greatest asset that went into the r0ket. Free labour is the capitalist way to put it, resulting in a product that generated business for various companies in the electronic business and that can be freely appropriated now that it ready for mass production without paying any licensing fees to the designers.

However, following Söderberg's argumentation, it is also possible to theorise the work of the r0ket team as a case of quasi-unalienated labour, a rather rare phenomenon in post-Fordist capitalism. The key point of such an argument is that the work did not happen in the context of labour relations and it found its meaning in the exercise of the anthropological essence of humans — creative and productive power. This brings us to subjectivation, that is, what kinds of subjects are produced through such a process. In the previous section I argued that the production process reflects the liberal subject which finds its satisfaction in individual expression in a twisted form that can be called collective expressivity. Here the argument is developed referring to alienation.

The central sight for the analysis of alienation in the context of capitalist production for many Marxists is the assembly line where the mass worker is locked into the time of the conveyor belt and doomed to perform robotic work according to machine logic. The organisation of the factory is crucial in that analysis, structured for the mass production of consumer goods. Clock-time is instrumental in such system to ensure the optimal productivity of the workers. Such a theoretical construct of the factory becomes relevant when it turns out that 3000 first generation r0kets are

arriving to the MuCCC hackerspace and they will have to be assembled in the confined of a limited time frame. While most parts are already on the PCB, some of them will have to be soldered on, and the packaging is also to be done. r0ket team sends out an open call through mailings lists of the MuCCC and the CCC organisation, inviting volunteers to play a part in the assemblage.

Participants describe the effort as epic, with every available horizontal and vertical space filled with circuit boards, and maybe around a 100 people spending more or less time with the r0kets soldering iron in hand. Reportedly, volunteers are travelling from another cities in Germany to join the mission. It is an opportunity for many to check in to MuCCC for seeing friends and acquaintances, and for other to check out the hackerspace and learn soldering. The process happens in the MuCCC, taking advantage of three defining characteristics of hackerspaces: (1) readily available workshop space and soldering equipment, (2) 24/7 access for members, (3) racks of Club-Mate available at the premises. The last may require further explanation: Club-Mate is a carbonated beverage made from mate tea leafs with some caffeine content, from a small brewery in Münchsteinach, Germany. It has a cult following in the hacker scene, and it is used to keep hackers alive and kicking throughout long sessions. All in all, core members still has to put in many extra hours bending over the soldering iron to finish on time, but finally all r0kets come together for the first day of the Chaos Camp. One person proudly says that "I have probably had each of them passing through my hand." (pers. comm.) Tellingly, since workers are not payed, and volunteers come and go as they please, there is no way to estimate the clock time — more precisely: the abstract labour — that went into this phase of the r0ket production. As Raymond writes in *How to Become a Hacker* (2011), "Being a hacker is lots of fun, but it's a kind of fun that takes lots of effort. The effort takes motivation." The moral is that even a typically factory process can be organised according to a less alienated community-based model, although not without stretching the limits of the hackerspace milieu. The repetitive work involved in

assembling a great amount of r0kets is against the hacker spirit. After all, this is why machines were invented in the first place. Raymond adds in the same text that "No problem should ever have to be solved twice." Consequently, the next generation r0ket, as we have seen, gets outsourced to China where the company has a factory infrastructure.

So far the emphasis has been on the production and distribution of the r0ket. Looking over the use cases of the r0ket covered above it seems that they are often concerned with play, more or less explicitly. INVADERS, the Space Invaders game in the default firmware, already laid the ground, and later expanded to a two player version with two r0ket, just to go massive with at the Congress with more than hundred people controlling two rackets. The laser tag game built on more sophisticated technology and involved building a whole appliance around the r0ket, but the whole point was to get people running around chasing each other with makeshift toy guns. However, even considering the Geiger counter workshop, it is safe to assume that most attendants have never used their counters in a serious situation, so it was a fun project for them as well.

While Söderberg argues that hackers do "play struggle" against capitalism, along with Himanen who also insists on the playfulness of hackers in the face of the Protestant Ethic, they only refer to developing useful software applications as a mix of work and play. They fail to show hackers engaged in actual play. Here we have proof that hackers are not only playful but they are actually developing and playing games, both simple and complicated ones.

I find this extremely important to point out because while in general I agree with Söderberg that "play relates to wage labour as negation", I also think that play has been thoroughly co-opted by capitalists in recent years to become a productive activity for capital accumulation. Many work processes have been reorganised to be more engaging and rewarding and social for the workers. Gamification is becoming a term in self-development and managerial literature (McGonigal). Companies like the Gamification Initiative at SAP help enterprises to "Make [employees] Work

more Fun" (see "Gamification facts and Figures", *SAP Gamification Initiative*). In light of this it is important to see that while hackers also learn from games, they are often engaged in it just for the entertainment value. In fact "hacking for fun and profit" is a widespread slogan in the scene. It is also true that hackers exploit the freedom that comes from precarious work, freelancing and outsourcing, which often enables highly qualified workers to concentrate on their own self-development, choose their own tasks, and organise their commissioned work in their own way.

A recent example of a commercial project involving the r0ket which builds on the playful experiments before is the theatre play *Crash Test North City — Do My Games*, organised by Dortmund Theatre. Director Jorg Lukas Matthaei works with members of the local Chaostreff hackerspace to set up a series of games in various locations across the city. During one scene r0kets are given out to six participants who have to solve poem quizzes by ranking quotations from Goethe, Grillparzer, Wedekind and Söderberg's favourite game theorist Schiller. If they score, the lights and the music in the room change. It is an example of r0ket in a professional, commercial setting, but it also brings together many motives mentioned before about hackers. The r0ket is staged here in the intersection of aesthetics, play and technology: so many potential escape routes from the iron cage of capitalism.

To conclude, most hackers have no scruples contracting commercial companies or being contracted by commercial companies, yet they often manage to do this on their own terms, and once finished, take the time to pursue personal projects. Thus they often extend and repurpose capitalist processes, which is possible because of the flexibility brought by the post-industrial restructuration. In particular, their freedom is often based on the free time afforded by precarious work patterns — most of the activities above are pursued without payment, yet hackers are typically not very poor. The profile of the perfect capitalist subject — self-managing, self-programmable, self-motivated and eternally productive — largely overlaps with the hacker ethos, yet the latter contains disjunct elements that go further than the outlines of the former.

These findings complement previous research which is mostly looking at hackers when they are writing software code which is directly useful for the industry, no matter if they are payed or not. The electronics projects in this paper have no direct commercial application, although they can be considered experiments or prototypes which can be transformed into products. Thus it becomes evident that the playful element is not merely an inherent flavour colouring serious work assignment, but actually manifests itself in what any observer would call games.

Furthermore, such elusive attitude to capitalism is in direct opposition to the norms of anticapitalist social movements, who usually refuse to do anything with commercial companies. This is all the more striking since these social movements often style themselves similarly to hackers. Importantly, the hacker strategy which is not confined by the strict principles of the social movements seems to be more productive in many cases.

Modernity

In this section I mostly use the concept of modernity as a shorthand for modern science. Since modern science plays a significant role in the construction of truth tests (Boltanski 2009) in modern society, this has wide-ranging implications.

Himanen identifies the ethos of the academia as similar to the hacker ethos but in contrast with the ethos of the monastic order, which in turn defined the essence of the Protestant Ethic of capitalism. Söderberg follows Himanen's argumentation that hackers are differentiated from other workers because they are driven by play and passion. Raymond already presents how the concrete organisation of work around these principles can look like when he describes his experience in free software development and outlines the critique of proprietary software development practices. Latour's early ethnographic work (1986) emphasises that far from the ideals of modern science, the facts born in the laboratories are negotiated in networks between

institutions, instruments and scientists themselves. Latour calls for the recognition of his findings to develop a more accurate perception of science, and in turn a more accurate science. Improving on this, Pickering points out that his cyberneticians were already doing brand of nonmodern science which goes beyond the concept of knowledge as a factual representation.

Here I develop this line of thought further to see how hackers are doing a science which differs markedly from the idea of modern science and modern scientific management, using the ethnographic material gathered about the r0ket device as a starting point. It is important to keep in mind that much of what all the aforementioned scientists and pseudo-scientists are doing is more or less based on modern science. The importance of these investigations is not simply to set up a dichotomy but rather to shed light on how the logic of modern science is interrupted by certain practices in specific social milieus much more than in the traditional corporate or academic research laboratories.

My particular argument here is that hackers are more likely to engage in scientific work which strays away from the idea that knowledge is a representation that has to stand in a correct relationship to reality. These kinds of activities go beyond that correspondence in their various ways. Initially, I concentrate on the purpose and hardware design of the r0ket, later looking at the various software applications installed on the r0ket in the default firmware and how they point to further interesting practices. Finally, I will consider the production and development process of the r0ket as well as its use cases, highlighting their spontaneity.

Amongst other things the r0ket is called a development board for rapid prototyping. Rapid prototyping is used as a term in physical design, electronic construction and to a lesser extent in software development. In physical design it means using a 3D printer or similar machine to make

a small-scale model of the proposed design in order to evaluate it in its physicality rather than on the drawing board. Hackers have used it to print actual useful objects, such as cases and boxes for the r0ket device. 3D printers useful for this kind of rapid prototyping are a staple of hackerspaces.

However, the r0ket is used for rapid prototyping in electronic design. This type of rapid prototyping means starting construction without a design, and only a rough idea of the goal. Furthermore, it involves using general purpose parts and loose connections which are not necessarily suitable for production use or mass production. It usually involves some backtracking and a number of iterative prototypes, each of which builds on the previous one. It is a specific form of development that is particularly suitable for the manner of working lauded by Söderberg as a germ form of unalienated labour, by Himanen as an antidote to the metal cage of capitalist rationality and by Raymond as a more efficient industrial mechanism. As noted above, all of them agree that this activity is guided by play and passion. Since during rapid prototyping is not the execution of a plan, the steps to be taken are often unforeseen, and the results surprising. Moreover, it is distinguished by its speed, producing practical results much faster than a planned process. Naturally, the results tend to be rough and unpolished, and evidently the prototype is far from the product itself. It is easy to see the affinity between play and rapid prototyping, and indeed, hackers often engage in the process for the sake of it.

A development board like the r0ket facilitates this approach to electronic construction through its deterritorialised design. On the one hand, it presents the users with a wide variety of pre-installed parts, like the light sensor, the display, the USB connector, the battery, LEDs of various colours, and so on. These are the most common components in ad-hoc electronic work and sessions often begin with connecting these to each other — the r0ket short-circuits that process by presenting them in a preconfigured assemblage. On the other hand, all these are easily reached and utilised through the processor. Without the r0ket, once a microprocessor is connected to these electronic parts, a further microprocessor programmer device has to be connected to perform the

programming. In contrast, the microprocessor in the r0ket can be programmed using a simple laptop and a USB cable. Of course, not all problems can be solved in such a way, but in a number of cases such an approach can yield faster results that are more adapted to the use case. Just today (May 12th, 2012) I heard feka from the Hungarian Autonomous Centre for Knowledge refer to such problems as "easily prototypable". His definition is that an easily prototypable problem can be solved faster through trying it out in practice than thinking it through in theory. Naturally, this ratio is highly dependent on the researcher's skills and motivations for theoretical analysis versus rapid prototyping.

The analogue of rapid prototyping in software development is exploratory programming, which is also distinguished from other software development methodologies by the lack of a prior specification and the fact that the final result is judged based on its practical fulfilment of its goal rather than its correspondence to the specification outlined (Green & DiCaterino 1998). A slogan for such rapid prototyping could be a quote from an overheard conversation at the Chaos Communication Congress. "Two hours of planning can save you two days of [software] coding.", proposed one participant, echoing the traditional wisdom of project management. Another hacker retorted that "Still, days of coding can save you two hours of planning." The point of this joke is that since hackers are passionate about coding, they are willing to do more of it to avoid the boring parts of the process. However, it also echoes the emphasis on practice rather than representation, described by Pickering, which characterises both the various types of rapid prototyping.

The default software bundled with the r0ket is a collection of some activities that inspire hackers, and therefore worth taking into account. For instance, PWGEN, the password generator software which is intentionally broken is a security tool. Hackers are associated with computer security in the popular imagination, and not without any grounds. According to my field work, while people who are working with the r0ket are often university students studying computer science or similar

disciplines, when they have a job they are usually programmers and system administrators with an unusually high frequency of them focusing on security. The dream job in this category is penetration testing and software auditing. Penetration testing means trying to break into a system with the permission of the owner in order to see if it really secure. Penetration tests usually find a number of "security wholes" in systems that are theoretically secure. Similarly, a software audit means that the programmer tries out and reads through the source code of a program in order to verify that it actually works as advertised. Both activities take the notion that what is theoretically sound is not necessarily sound in practice. In other words, these exploit the gap between knowledge as representation and performance as practice. To quote Alfred Korzybski, "the map is not the territory" (Pula 1994:xvii). In the case of PWGEN, its author intentionally designed the defect, presenting an interesting find for those who audit the software — and a source of security mistakes for the vast majority who do not. Eventually, he revealed the secret, and called on his peers to perform penetration tests to check if anybody actually used these defective passwords on the computers connected to the Camp network. While this practical joke had many other interesting cultural aspects explained in the previous section, it also used this little electronic gadget to make many opportunities for the participants to engage in security research — which most hackers love to do. Of course PWGEN generates only 8 character long passwords anyway, which nobody would use for securing anything serious (in theory), so the whole story was not a serious security threat to the community, only a practical joke with a political and professional point.

Another default program installed on the r0ket is MANDEL, a fractal viewer that displays a Mandelbrot set. Hackers love fractals because fractals have infinite resolution, complexity and self-similarity despite the fact that they are generated by a simple algorithm. They were also the first universally accepted instances of beauty which are generated programmatically through a mathematical algorithm, proving the point of the hacker ethic that "You can create art and beauty

on a computer" (Levy 2012:31). To write a small computer program that brings something that has infinite resolution on the tiny 96×68 pixel screen of the r0ket is a statement in itself, a performance to prove that the clever design of seemingly primitive things can engender great complexity. Visually, a fractal viewer is perhaps as close as one can get to the challenge of representation, since a fractal can never be fully represented — despite the fact that the best way to grasp it is exactly a representation. Of course it is performance which resolves this paradox: fractals are explored in time interactively through fractal viewers that can generate new images of them depending on the desired zoom or pan direction.

Fractal explorations have been a common feature of demoscene productions, the short animations generated by computer programs with which hackers competed to push the limits of the earliest personal computers. It is in this context that the comment from the author of the MANDEL application can be best interpreted, when he argues that this piece of software showcases the superior CPU power of the r0ket device by showing what it is possible to achieve with it. The immense creativity and technical virtuosity that went into demoscene animations can be attributed to the radical simplicity of these primitive machines, enabling many teens to understand their operations bit-by-bit. While this is not possible any more with today's PCs, the r0ket brings back these technical possibilities — and aesthetic experiences — to a new generation of hackers.

Finally, INVADERS is a re-implementation of the classic computer game. It is rather obvious that games provide an unplanned and unpredictable experience to their users based on a performative experience. One might ask what is nonmodern in all these software, especially if we accept that any kind of scientist is generally researching unknown topics and often surprised by the results. It could be also difficult to grasp this without experience in the "industry". The answer is that throughout its short history, the expectation for computer scientists has increasingly been that they should create logical machines that worked reliably and repeatably,

often according to a plan developed in another department with less grasp of technicalities. In contrast with that, the whole point of the above applications is to make the man and the machine to search for unpredictable patterns, be they "random" strings of characters, bugs in the software code, aesthetic or mathematical patterns, or simply tactical situations with intensity. While Raymond argues that hackers can write software that work as well as corporate creations (or even better), I try to show here how hackers prefer to write software that works differently.

While the attention to emergent phenomena has been demonstrated in the software and hardware practices above, it is also instructive to see how such patterns work in the organisation of the work. The previous chapter describes in detail the complicated process of producing, distributing and utilising the r0ket for various projects. Here I would like to highlight the fact that there has never been a precise business plan worked out for these phases in the life of the r0ket. For instance, it was not clear and there was no way to know if the team can mobilise enough people to assemble the first generation of the device on time, or — as they admitted during the r0ket presentation at the Congress — the choice of the company for printing the circuit boards happened in a rather ad-hoc manner following the recommendation of an experienced hacker, and there was no time to "shop around" for the best offers. The fact that all these went without major setbacks is a testimony more to the solidity of the social and infrastructural network surrounding the r0ket from the moment of its inception rather than the result of methodological planning and calculation. It is particularly puzzling that during my field work and interviews I found no concept about the future of the r0ket — in fact, team members were happy to release the second generation r0kets at the Congress and they were also happy to forget about the whole project for a while. These small organisational notes show a high reliance on the community and the hope that once something good is "out there", people will take it up and it will have a life of its own. The whole ambition behind making the r0ket is arguably to see how the audience can use it in yet unknown ways. Raymond already incorporates this attraction

to the emergent into his writing on hacker culture, but he does not work out its significance systematically: "Any tool should be useful in the expected way, but a truly great tool lends itself to uses you never expected."

On the other hand, there is no denying that both r0ket team members and many r0ket hackers make use of modern scientific and industrial methodology, knowledge and practices in electronics and software development. They effectively build on their studies as students and their professional background, merging them with their experience in tinkering in the context of the hackerspaces scene. The processes converging in and around the r0ket are for the most part derived from the lineage of modern science. Nonetheless, the peculiarity of the r0ket is in its derivation from the mainstream of such lineage. My argument is that hackers mobilise modern science but with a peculiar twist which enables them to bypass some of its limitations.

Concluding remarks

Peer production goes physical

One research question concerned the establishment of the limits of peer production when it comes to the manufacture of physical artifacts. The production and use of the r0ket has clearly been a cooperative endevour undertaken largely outside the frameworks of individual invention, capitalist organisation of labour, and modern scientific practice — through a process that largely corresponds to previous descriptions of peer production. However, the ethnographic material shows two divergences from models of free software development outlined in the accounts reviewed above.

On one hand, it is clear that in order to organise the production of the r0ket, hackers had to engage with mainstream structures like commercial companies, established civil society organisations like the Club, and even factories to a greater extent than software programmers would ever have to do. Himanen, Söderberg and Benkler takes into account the role of commercial enterprises in free software production, but they find a much looser integration. This puts the paradox outlined in the literature review firmly back on the table: how to interpret the fact that although hackers exhibit behaviours that are often seen out of line with dominant models of subjectivation, production and scientific inquiry, they also incorporate these in their practices, while in turn hacker practices are routinely incorporated into these dominant structures.

On the other hand, it is evident that embodied communities and physical spaces are indisposable for the production of tangible artifacts on a cooperative basis. My surveys also show that the main site where the r0ket is utilised are the large-scale hacker gatherings when practitioners come together in a common time and space. As shown above, these are also the occassions when the radio frequency communication features built into the r0ket make the most sense. During these times the radio waves create virtual communication channels that complement the physical togetherness of participants in a way that can be theorised as a rudimentary augmented reality. Significantly, this territorialised and localised process even leads to the closing of at least some virtual communication lines — the invite-only development mailing list used by r0ket developers in 2011 is a case in point.

The incorporation of these elements which are relatively novel from the point of view of peer production theory — shared time and space, embodied communities, industrial production, seed capital, to mention a few — did not fundamentally block the flow of peer production processes.

Cybernetics

Cyberneticians looked for a way to design adaptive systems that can stay relevant in a fundamentally unpredictable world. They found that in order to adapt to its "essential variables", a system have to be able to exhibit emergent behaviour which is at least as diverse as the relevant variables. While they were trying to design adaptive machines, they discovered that complex systems can work according to relatively simple principles. A key element of such designs was the feedback loop, which adds information about the state of the system itself to its inputs.

Their work could not integrate into the circuits of modern science and many pursued their most important research in the form of pet projects. Pickering (2010:54-60) explains their failure to find a social basis for their work with their visionary outlook that questioned knowledge as representation, and preferred performance. These attitudes went against the established norms of modern science, which aims at producing knowledge as representation, and which is divided into disciplines — for example into departments on the university level — that facilitate that work.

The r0ket device does not do adaptation itself, but it is a general purpose device — even more so than the personal computer — that can be adapted by its users to a great variety of situations. This is largely due to their bus connectors through which they can interface with almost any other electronic device. Furthermore, r0kets create a wireless mesh network with each other. It enables them to pass messages to each other to create feedback loops, as we have seen in the case of the mass pong game. Though r0kets themselves do not adapt to their environment automatically, considered together as an assemblage with hackers using them they are surprisingly flexible, as the wide range of applications described above show.

Few companies would design a single end-user product for all these purposes. In fact, as Cory Doctorow (2011) pointed out in his presentation at the Congress entitled "The Coming War on General Computation — The Copyright War was Just the Beginning", there is a new trend

displacing the convergence of electronic devices that points to the direction of single-purpose machines, as evident in the success of ebook readers and iPads. In this context the exploratory programming, rapid prototyping and general tinkering going on in hackerspaces is rather out of place. No wonder that the similarity to the creations of the cyberneticians is so striking. The example of the Musicolor machine was already mentioned above: a synaesthetic piano which went from the theatre to children's toys and ended up in the business of teaching typewriting. What is more, another Pickering subject Grey Walter's electronic tortoises started in the same way as an afternoon pastime on the kitchen table. Eventually, they resulted in medical publications about brain science, became the favourite of television shows, and ended up laying the ground for the emerging field of robotics. Finally, Ross Ashby in his futile design for a brain, the DAMS machine — a jungle of electrodes that never worked — embraced the principles of rapid prototyping. As the complexity of the machine grew, he realised that he will never be able to come up with a blueprint for the machine. He has to proceed without an exact plan, continuing the research on the basis of the little knowledge that he had, complemented by his intuition and the trial and error process of trying out different solutions in practice. He wrote in his journal that "One is almost tempted to dogmatise that the Darwinian machine is to be developed only by the Darwinian process!" (Pickering 2010, 127)

During the course of their lifetime r0kets have developed multiple identities, starting as a learning project and going from a one-off conference badge to a full-fledged ARM development board, from souvenirs of a summer camp to collector's items, little artworks in themselves, and in certain circles, signs of distinction. In this paper they are the encyclopedia of hacker culture and as argued above they may have a future as the perfect mass manufactured product for electronic enthusiasts.

To use Latour's terminology for a moment, this is all made possible by the r0ket's ability to forge cultural and technical connections with people and other devices. Or to advance on Latour's idea of groupings: while the r0ket can be considered a grouping in itself, its various parts like a light sensor or a low-resolution display can all be used separately, or replaced with other hardware. Similarly, the r0ket can easily be incorporated into a larger group, involving for instance the components of a laser gun. These properties could be conceptualised in the term "unstable grouping". Maybe Deleuze and Guattari's terminology is more enlightening, with which we can say that r0kets are devices criss-crossed with lines of flight, lines of deterritorialisation which open up the possibilities for creative reterritorialisations.

The social basis of cybernetics functioned similarly on the macro level: while cyberneticians often worked on their most important projects on the kitchen table, in the larger context of the society they found many disparate outlets. According to Pickering the central hub of British cybernetics was the Ratio Club — a dining club for discussion which can roughly be consigned to the category of civil society. Then, the first British cybernetician Stafford Beer made a living much of his life as a business management consultant, running his own little company in the private sector. Finally, Ashby and Walter worked in hospitals, psychiatric hospitals and universities. Pickering notes that all of these were ad-hoc solutions that could never fully accommodate the research interests of cyberneticians, and points to other extra-institutional and transversal constellations, produced by the sixties counterculture as the self-organised Free University in London, or the Fun Palace, a plan for an adaptive learning centre on the bank of the Thames. He argues that these were surfaces of emergence for practices based on the cybernetic ideas — sites where its logic is put to work thanks to the cultural influence it had, or the direct involvement of cyberneticians. I argue that hackerspaces which make possible the r0ket are comparable configurations, with a similar ambiguity as to their place in the taxonomy of the institutions of modern society. They cannot be assigned to the private sector because they are not profit

oriented, despite there being some profit oriented activities going on in them and they occasionally engender spin-off companies. They are not accredited universities or other state-sanctioned organisations, even though they are often studied as peer-learning or informal learning environments (Hunsinger 2011). Finally, while they are often run as an association or foundation, they are not the typical Non-Governmental Organisation either, since they are not providing any formal services to the general public, nor working on a specific social cause or problem. Indeed, some of them are not legal entities at all. Indeed, it is exactly this malleability of these community-run quasi-institutions which can nourish exploratory projects like the invention and usage of the r0ket device. It is in these strange places where Pask's Musicolor, Walter's tortoises, or Ashby's DAM would have found their social basis and appreciative audiences, where they would not have felt out of place.

However, there are significant differences as well, the most important two being that hackerspaces are relatively *successful* and they represent a *popular social practice*. I argue that there is a connection between the two. While the cyberneticians in Pickering's presentation appear to be a troupe of lonely geniuses, the hackerspaces cater for existing working groups and communities, just like the one in MuCCC which housed the r0ket team. When Pickering writes about Gordon Pask and Stafford Beer working on biological computers, he notes that whatever the potential was there in the research project, it could not be brought to fruition in private homes and without institutional support. Indeed, the r0ket is in some ways an exceptional creation of hackerspaces coming from the hinterland of the European scene. As I described above, the r0ket team could count on the moral, financial and infrastructural support of one of the world's largest hacker organisations, with a history going back to 1981, which routinely draws several thousand participants to its regular gatherings. This is possible because cybernetics has been a twisted and sidelined scientific discipline, while hacking has managed to develop into a major yet still twisted technological subculture. A potential reason for this, as I tried to demonstrate in the previous

sections, is that while hacking can be seen as a departure from modernity in general and modern science, capitalism and liberalism in particular, it still managed to build on the foundations of these. More precisely, hackers found a way to incorporate the ruling systems into their own and to establish productive connections more than the cyberneticians did. This is not to say that cybernetics did not manage to make connections with other scientific disciplines: indeed, in its capacity as a general science it did so more profoundly and richly than any other field (with the possible exception of its predecessor, statistics) in the twentieth century. However, as Pickering argues it dissipated in these connections and failed to find a social basis in which to ground itself. It is interesting to see for the future what happens to hacking as it currently expands to areas beyond computer science to industrial design, cooking, knitting or genetics.

However, what is more important is to look at the common thread that binds these practices together. In the case of British cybernetics, it was an ontological outlook that posited a fundamentally unknown world. When scientists connected adaptive systems to each other through feedback loops, they observed unpredictable behaviours which pointed towards an emergent order — a certain drive for self-organisation. Hackers do not necessarily concentrate on researching emergence per se, but base their work on the nonmodern conclusions of cyberneticians. The bottom line is that given the right configuration of a network with enough "free radicals" (open network nodes), productive patterns will emerge. The collective authorship described in the section on liberalism is not necessarily meaningful in the sense of representation, since "r0ket team" or "Chaos Computer Club" does not point to a stable authorship, but merely a moniker for the performative network of people which produced the r0ket. This is why there is no formal membership in team r0ket except implicitly through actual work done — through performance. In the section on capitalism I have shown how hackers combined capitalist and non-capitalist resourcing like commercial contracts, gifts, sponsorship, volunteering, zero interest loans, and so on to produce the r0ket. They had few ideological doubts about sources, as long as

they contributed to the construction productively. Furthermore, there was no business plan for the future behind the r0ket, other than the belief that if it has enough functionality that users can utilise, it will find its applications. Indeed, people used it as an opportunity to hold workshops for a small charge, to incorporate r0kets into their for-sale Geiger Counters, and to use it as props in a theatre play. Finally, the previous section on modernity was mainly used to show how emergent patterns receive special attention from hackers on the level of programming code.

A concrete example of the affinity between classic cyberneticians and hackers with r0kets is the mass pong game at the Congress involving some hundred participants who controlled the two pads on the big screen collaboratively through their badges. Loren Carpenter conducted the same experiment with several hundreds people in California during 1991 using analogue controllers. His point was to prove that a multitude can act cooperatively without central control if enough feedback is provided.

The free radicals or free variables that are built into the r0ket, as well as the hackerspaces in particular and the hacker scene in general provide enough opportunity to follow haphazard paths whose destination cannot be mapped out adequately, only explored in practice. It allows the explorer to trace the clues that arise from the material (technology) itself during the development process, an ability which is severely limited in modern institutional contexts. For example there would have been no institutional support or scientific rationale to develop such things as a laser tagging system, a dating application or a mass computer game (pong) for the r0ket, but the hacker gatherings with their embodied communities were great environments and the development board could easily accomodate these emerging needs.

No wonder that one of the new hackerspaces in the Netherlands is called Terra Incognita, and the 26th Congress (in 2008) sported the slogan “Here Be Dragons” — the translation of Hic Sunt Dracones, a phrase which marked uncharted territories on medieval maps. Such a relation to

technology and nature is what Heidegger calls the poetic revealing, as opposed to the technocratic domination of nature through modern technology by enframing it. Supporting evidence is the vast amount of cultural meaning incorporated in the software and hardware design for the r0ket, which is exemplified by the fact that the description of the r0ket device almost unfolded to a full scale ethnographic description of the whole hacker scene from the knitters to system administrators. Instead of creating a mere microcontroller, r0ket developers and subsequent users incorporated their whole hacker life into this gadget. As Heidegger notes in connection with Hölderlin when he seeks the essence of the poetical, the eminent poet calls forth a whole people in his creation.

Excess

All in all, what is the relationship of hackers to liberalism, capitalism and modernity? As shown above, no simple opposition can be posited. The best way to formulate it is that the relationship of hackers to liberalism, capitalism and modernity is that of excess. They bypass the limitations of these categories by growing out of them. In fact while Himanen, Söderberg, Raymond and Benkler are caught up trying to theorise why people would pursue projects which they find enlightening, fun and profitable, they also point out how restrictive frameworks like my trial categories block the incredible productivity inherent in free pursuits and voluntary cooperation.

More precisely, such excess can perhaps be best grasped in terms of Bataille's general economy. He writes about excess in his three volume *The Accursed Share* (1991), where he proposes that all life is characterised by an irrational abundance of energy, in contrast with the view of restricted economy which works with rational subjects in a situation of scarcity. Life forms have to get rid of this excess energy in one of three ways: through growth, luxury or conflict. The latter two arises when the limits of growth are reached.

I argue that hackers often manage to bypass the limits of growth built into expressive individualism and restrictive intellectual property rights based on authorship, hierarchical management structured based on capital accumulation, and modern science and technology based on knowledge as representation. While doing that, they may turn to a playful luxury which can potentially route around the limitation, or a conflictual sabotage which might break it. In any case, they are driven by a spontaneous life energy that individualist liberalism, exploitative capitalism or rational modernity cannot fully capture.

Þ

References and Bibliography

Agamben, Giorgio. 1996. Beyond Human Rights. In *Radical Thought in Italy: A Potential Politics*, edited by Paolo Virno & Michael Hardt, 159-167. Minneapolis, MN: Minnesota University Press.

Ahmad, Usman. 2005. "Great Programmers Answers….Interview with Steve Yegge, Linus Torvalds, Dave Thomas, David Heinemeier, Peter Norvig, James Gosling, Guido Van, Tim Bray." *Life Beyond Code* (blog). https://usmanahmad.wordpress.com/2006/08/02/great-programmers-answersinterview-with-steve-yegge-linus-torvalds-dave-thomas-david-heinemeier-peter-norvig-james-gosling-guido-van-tim-bray/

"Arduino UNO SMB Edition" (item on sale), *Pollin Electronic*, accessed May 28, 2012. http://www.pollin.de/shop/dt/NzE4OTgxOTk-/Bausaetze_Module/Module/ARDUINO_UNO_SMD_Edition.html

Barbrook, Richard. 2007. *Imaginary Futures: From Thinking Machines to the Global Village*. London and Ann Arbor, MI: Pluto Press.

Bataille, Georges. 1991. *The Accursed Share, Volume 1: Consumption*. Translated by Robert Hurley. New York: Zone Books.

Bernard, Harvey Russel. 1995a. Participant Observation. In *Research Methods in Anthropology*, 136-164. Walnut Creek, CA: AltaMira Press.

Bernard, Harvey Russel. 1995b. Unstructured and semi-structured interviewing. In *Research Methods in Anthropology*, 208-236. Walnut Creek, CA: AltaMira Press.

Blanc, Sabine and Ophelia Noor. 2011. 30 Years of Political Hacking. *Owni.eu — News, Augmented*, 30 September. http://owni.eu/2011/11/08/30-years-of-political-hacking/

Boltanski, Luc. 2011. Political Regimes of Domination. In *On Critique*, 116-149. Cambridge & Malden, MA: Polity Press.

Boltaski, Luc and Eve Chiapello. 2005. *The New Spirit of Capitalism*. New York, London: Verso.

Bonnington, Christina. 2012. Probe Finds 'Serious and Pressing' Violations at Foxconn iPlants. *Wired*, 29 March. http://www.wired.com/gadgetlab/2012/03/apple-foxconn-audits/

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.

Briggs, Charles. 1983. Listen before you leap: toward methodological sophistication. In *Learning How to Ask*, 93-111. Cambridge: CUP.

DENIC. 2010. Whois information on r0ket.de.

Derrida, Jacques. 1992. *Given Time I. Counterfeit Money.* Chicago, IL: University of Chicago.

"DIY Projects, Inspiration, How-tos, Hacks, Mods & More @ Makezine.com - Tweak Technology to Your Will" (about page), *Make Magazine*. http://makezine.com/volumes/index.csp

Doctorow, Cory. 2011. The coming war on general computation — The copyright war was just the beginning. Presentation at the 28th Chaos Communication Congress (28C3) "Behind Enemy Lines", annual meeting of the Chaos Computer Club, Berlin. https://events.ccc.de/congress/2011/Fahrplan/events/4848.en.html

Forlano, Laura. 2012. Digital Media Research. Course at the Department of Political Science, Central European University, Budapest.

"Gamification Facts & Figures*", *SAP Gamification Initiative*, accessed May 28, 2012, http://enterprise-gamification.com/index.php/facts

Gilman, Robert. 1983. "Structural Violence Can we find genuine peace in a world with inequitable distribution of wealth among nations?". *In Context — A Quarterly of Humane Sustainable Culture* 4. http://www.context.org/ICLIB/IC04/TOC04.htm

Green, Darryl and Ann DiCaterino. 1998. "A Survey of System Development Process Models". *Center for Technology in Government*. http://www.ctg.albany.edu/publications/reports/survey_of_sysdev

"Graphs of r0ket/r0ket", *Github.com*, accessed May 28, 2012. https://github.com/r0ket/r0ket/graphs

guest. 2011. "there's no security in trusted boot or how I hacked 3000 hackers ;)", *Pastebin — #1 paste tool since 2002*. August 14th. http://pastebin.com/04wAcXzJ

Hammesrley, Martyn and Paul Atkinson. 1983. Research Design: Problems, Cases and Samples. In *Ethnography: Principles and Practice*, edited by Hammersley and Atkinson, 23-53. London: Routledge.

Hardin, Garett. 1968. "The Tragedy of the Commons". *Science* 162 (3859): 1243–1248.

Hardt, Michael and Antonio Negri. 2004. *Multitude: War and Democracy in the Age of Empire*. New York: Penguin Press.

Harvey, David. 2005. *A Brief History of Neoliberalism*. Oxford: Oxford University Press.

Heidegger, Martin. 1993. The Question Concerning Technology. In *Martin Heidegger: Basic Writings from "Being and Time" (1927) to "The Task of Thinking" (1964)*, edited by David Farrell Krell. San Francisco: HarperCollins.

Himanen, Pekka. 2001. *Hacker Ethic: A Radical Approach to the Philosophy of Business*. New York: Random House.

Hunsinger, Jeremy. 2011. The Social Workshop as PLE: Lessons from Hacklabs. In *Proceedings of the The PLE Conference 2011*, 10th - 12th July 2011, Southampton, UK.

Igoe, Tom and Dan O'Sullivan. 2004. *Physical Computing: Sensing and Controlling the Physical World with Computers*. Boston, MA: Premier Press.

June, Laura. 2011. 5 Minutes on The Verge: Limor Fried. *The Verge*, 23 November. http://www.theverge.com/2011/11/23/2583441/5-minutes-on-the-verge-limor-fried

Kellermann, Benjamin & klobs. 2011. NPC: Nerds' Pissing Contest — Mein Ruby ist besser als dein urxvt!. Workshop, 28th Chaos Communication Congress (28C3) "Behind Enemy Lines", annual meeting of the Chaos Computer Club, Berlin. https://events.ccc.de/congress/2011/Fahrplan/events/4722.de.html

Latour, Bruno and Steve Woolgar. 1986. *Laboratory Life: The Construction of Scientific Facts*. Princeton, NJ: Princeton University Press.

Latour, Bruno. 1993. *We have never been modern*. Cambridge, MA: Harvard University Press.

Lessig, Lawrence. 2004. *Free Culture: How Big Media Uses Technology and the Law to Lock Down Culture and Control Creativity*. http://www.jus.uio.no/sisu/free_culture.lawrence_lessig/portrait.letter.pdf (accessed May 28, 2012)

Levy, Steven. 2010. *Hackers: Heroes of the Computer Revolution - 25th Anniversary Edition*. Sebastopol, CA: O'Reilly Media.

Maker Carnival (website). http://makercarnival.com/BringMeChina.html

maxigas, "Hacklabs and Hackerspaces — Trancing Two Genealogies," in "Expanding the frontiers of hacking," ed. Johan Söderberg and Alessandro Delfanti, special issue, *Journal of Peer Production* 1, no3 (2012). Forthcoming.

"Modul r0ket" (item on sale), *Pollin Electronic*, accessed May 28, 2012. http://www.pollin.de/shop/dt/ODE4OTgxOTk-/Bausaetze_Module/Bausaetze/Modul_rOket.html

Nordgren, Andie. 2010. "Soldering is easy - comic adaptation of Mitch Altman's soldering teachings." *Andie's Log* (blog). http://log.andie.se/post/397677855/soldering-is-easy

Part Fusion Electronics, "About", accessed May 28, 2012. http://partfusion.com/about/

Pickering, Andrew. 1993. "We Have Never Been Modern (review)." *Modernism/modernity* 1 (3): 257-258. http://muse.jhu.edu/journals/modernism-modernity/v001/1.3br_latour.html

Pickering, Andrew. 2010. *The Cybernetic Brain: Sketches of Another Future*. Chicago, London: University of Chicago Press.

Plato. 1972. *Phaedrus*. Translated and edited by Reginald Hackforth. Cambridge: Cambridge University Press.

Plato. 1999. *Theaetetus*. Translated by Benjamin Jowett, ebook produced by Sue Asscher. Project Gutenberg. http://www.gutenberg.org/dirs/1/7/2/1726/1726.txt

Plato. Apology. 2012. *The Apology*. Translated by Benjamin Jowett. Wikisource. https://en.wikisource.org/wiki/Apology_(Plato)

Pula, Robert P. 1994. Preface to the Fifth Edition. In Alfred, Korzybski: *Science and Sanity: An Introduction to Non-Aristotelian Systems and General Semantics*, Fifth Edition. Englewood, NJ: Institute of General Semantics.

Raymond, Eric Steven. 2011. *How To Become A Hacker*. http://www.catb.org/~esr/faqs/hacker-howto.html

Raymond, Eric. 1992. *The New Hacker's Dictionary*. MIT Press: Cambridge, MA.

Rigi, Jakob. n.d. "Peer to Peer Production and Advanced Communism: The Alternative to Capitalism". *Critical Studies in Peer Production*, 2. Forthcoming.

Robfitz. 2011a. "Laser Tag." *28C3 Public Wiki*. https://events.ccc.de/congress/2011/wiki/Laser_Tag

Robfitz. 2011b. "Laser Tag." *Camp 2011 Public Wiki*. https://events.ccc.de/camp/2011/wiki/Laser_Tag

Rogers, Richard. 2009. *The End of the Virtual: Digital Methods.* Amsterdam: Amsterdam University Press. http://govcom.org/publications/full_list/oratie_Rogers_2009_preprint.pdf

Sec. 2012. *A simple tool to play around with the openbeacon packet traces* (documentation). http://www.42.org/~sec/tracking/README

Styhre, Alexander. 2005. Ideology and the subjectification of the entrepreneurial self. *International Journal of Management Concepts and Philosophy* 1 (2): 168-173.

Söderberg, Johan. 2008. *Hackering Capitalism*. London: Routledge.

Taylor, Charles. 1992. *Sources of the Self: The Making of the Modern Identity*. Cambridge, MA: Harvard University Press.

Taylor, Charles. 2007. *A Secular Age*. Cambridge, MA: The Belknap Press of Harvard University Press.

The Mentor. 1986. The Conscience of a Hacker http://www.ghostwheel.com/merlin/businesslike/hacker.html

Warfare Monitor & Shadowserver Foundation. 2010. *Shadows in the Cloud: Investigating Cyber Espionage 2.0*. http://shadows-in-the-cloud.net/

Weber, Max. 1949. Objectivity in social science and social policy. In *The Methodology of Social Sciences*, 49-112. Glencoe, IL: Free Press.

Weber, Max. 2002. *The Protestant Ethic and the "Spirit" of Capitalism and Other Writings*. New York: Penguin Book.

Weyand, Tobias and Christian Buck. 2011. 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. https://events.ccc.de/congress/2011/wiki/Lightning_Talks

Wilding, Adrian. 2010. "Redivivius: On Bruno Latour's 'Political Ecology'." *Cosmos and History: The Journal of Natural and Social Philosophy* 6 (1).

lilafisch & Stefan 'Sec' Zehl. 2011. r0ket++: The CCC-Badge — Now you've got that r0ket thing. What to do with it?. Lecture at the 28th Chaos Commication Congress (28C3) "Behind Enemy Lines", organised by the Chaos Computer Club, Berlin.

pfletz. 2012. “Launching Rockets with r0kets”. Post on the r0ket soup, January 8th.

r0ket tr4cker — 28C3 R0ket Tracking. http://longcat.de/ccc/28C3-r0ket-tr4cking/track.html

r0ket wiki contributors. 2011. *start.* http://r0ket.badge.events.ccc.de/start?rev=1308087273

robert. 2012. “Laser Tag m0dul.” *Part Fusion Electronics*. January 31st. http://partfusion.com/2012/01/laser-tag-m0dul/

“USB Missile Launcher” (item on sale), *Getdigital — Your Geek Stuff Supplier*, accessed May 28, 2012. http://www.getdigital.de/products/USB_R

Figures

Appendix of figures.

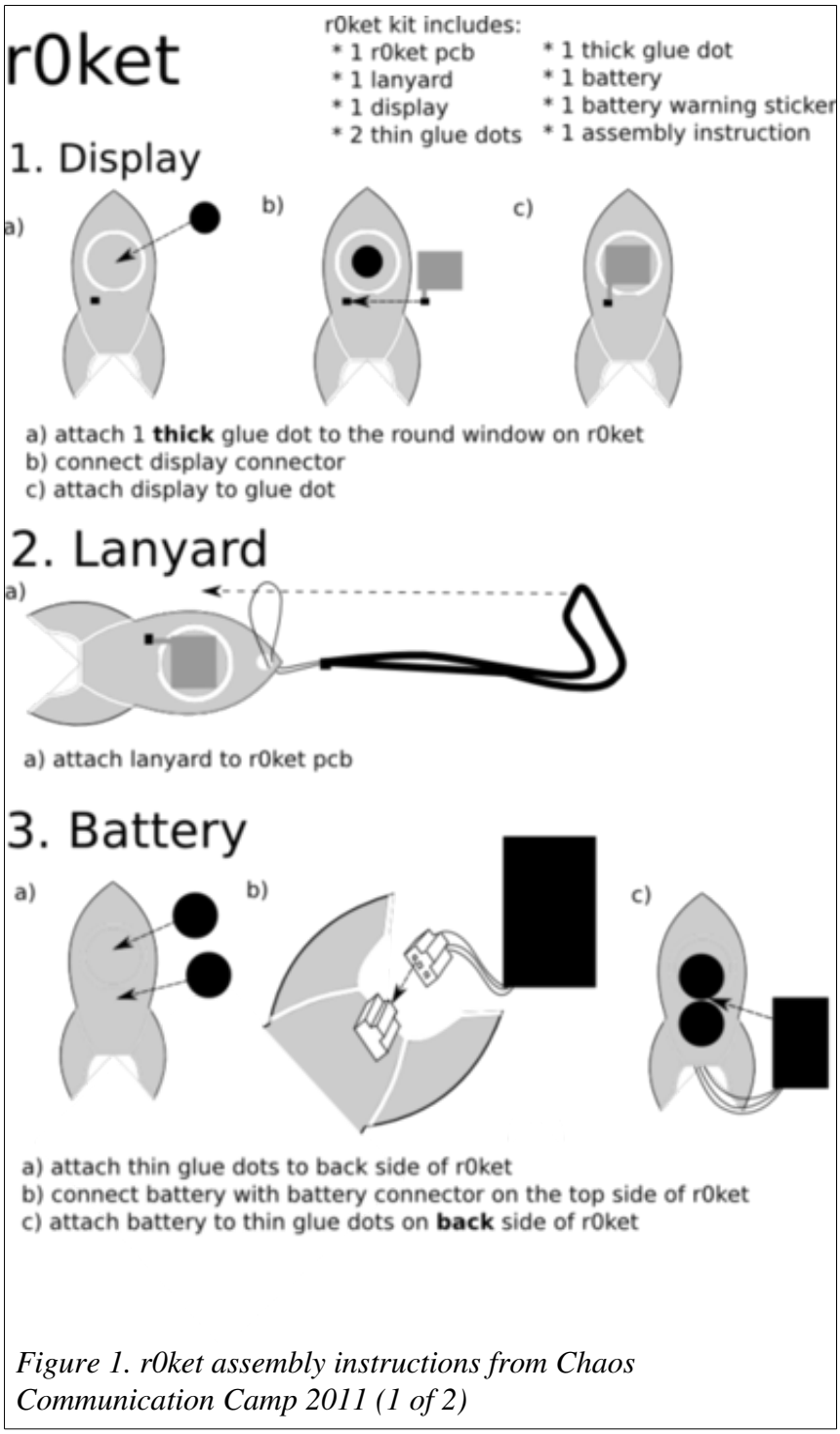

Figure 1. r0ket assembly instructions from Chaos Communication Camp 2011 (1 of 2)

Figure 2. r0ket assembly instructions from Chaos Communication Camp 2011 (2 of 2)

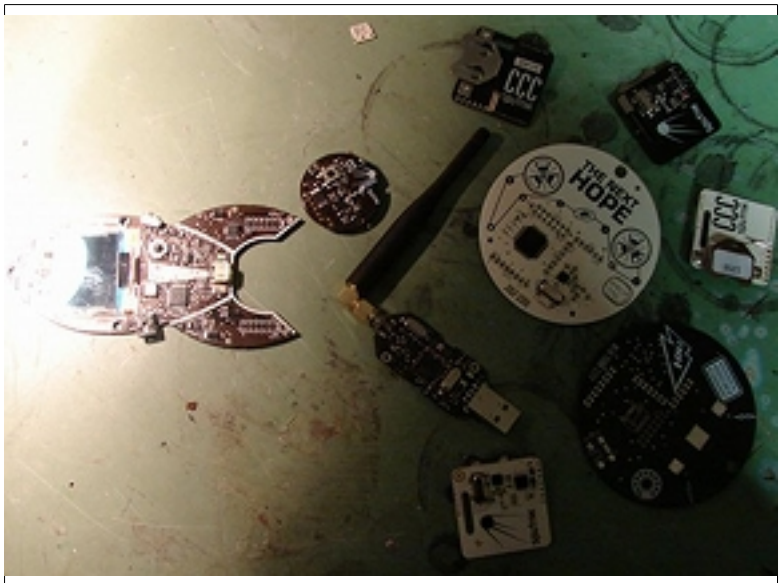

Figure 3. Badges from different hacker conferences

Figure 4. "m0re knittenart". from r0ket's soup, posted on August 14th, 2011.

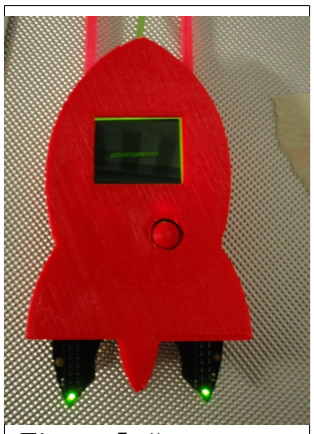

Figure 5. "reprap your own r0ket case!". from r0ket's soup, posted August 13th, 2011.

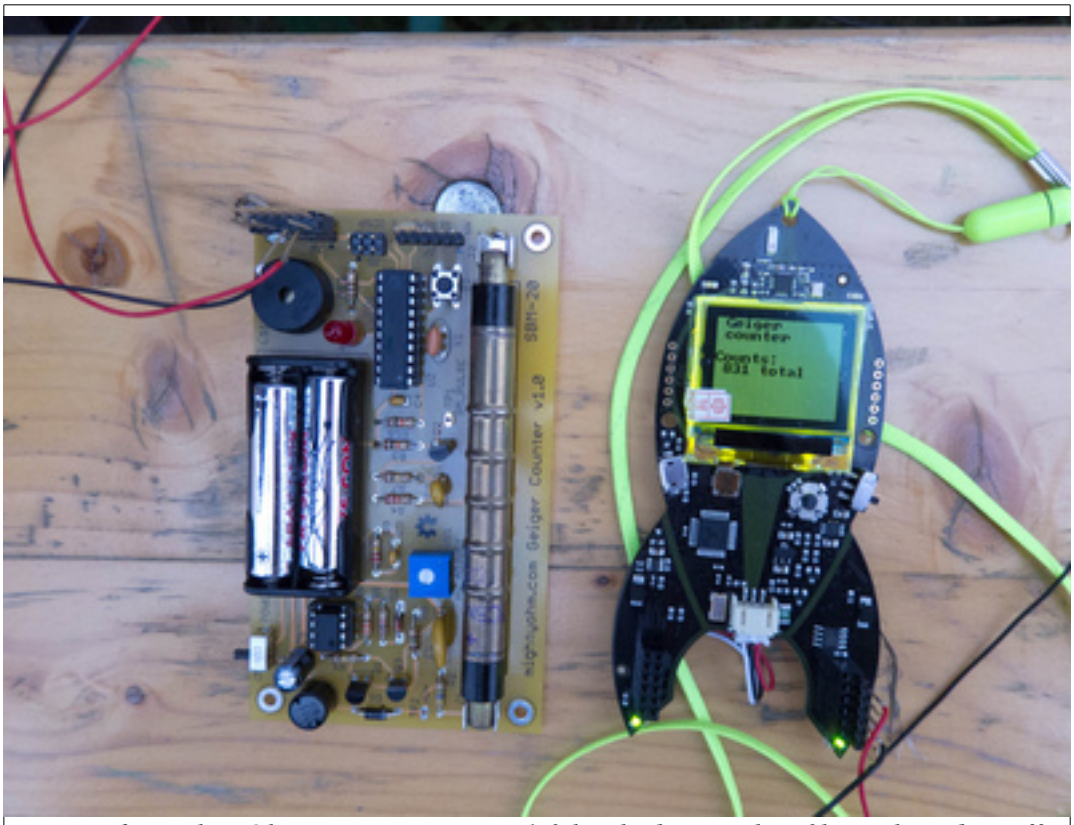

Figure 6. Mighty Ohm geiger counter v1.0 hooked up to the r0ket. Photo by Jeff Keyzer, license Creative Commons Attribution-Sharealike 2.0 Generic.

Figure 7. USB missile launcher, produced by DreamCheeky and donated to the r0ket team by the getDigital store ("USB Missile Launcher", Getdigital — Your Geek Stuff Supplie).

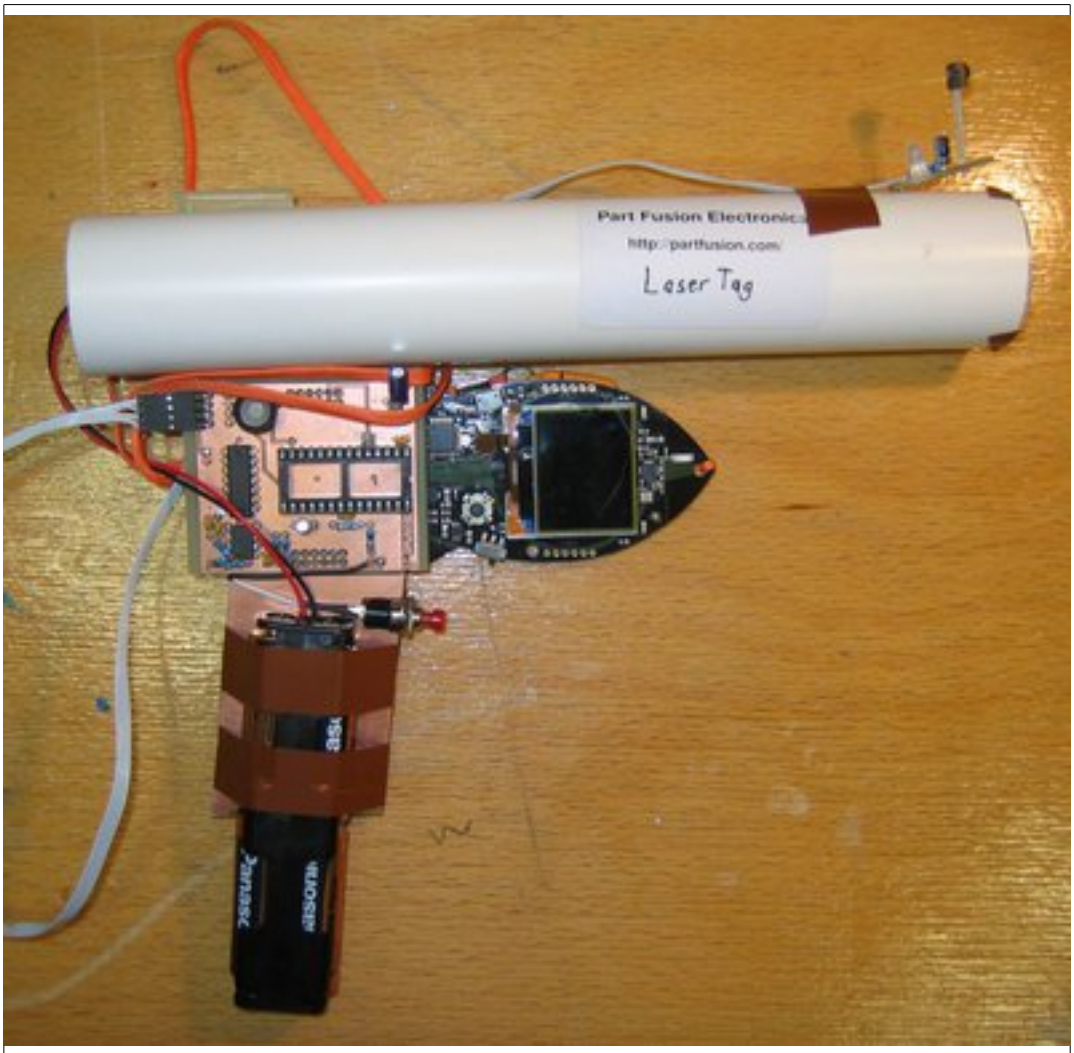

Figure 8. Laster Tag gun from r0ket and other parts by Robert Fitzsimons of Part Fusion Electronics.