Free Software and Anarchism
- does this compute?

Sandro Gaycken
Institut für Wissenschafts- und Technikforschung (IWT) - Bielefeld
Institut für Philosophie, Abteilung für Wissenschaftstheorie und Technikphilosophie - Stuttgart
Universität Bielefeld
sandro.gaycken@iwt.uni-bielefeld.de

Abstract: **The mode of production in free software development is often being described as anarchical. Despite this attribution seems not initially intended in any fundamental political sense, this sense starts to transfuse the discussions. This invites to a closer look at the reference: what it is, what it's not and what it could be. And once viewed from general anarchist theory and the anarchist theory of technology, any political relation seems to vanish. But despite this first stance, a demonstrative value can still be obtained as soon as some critical remarks are acknowledged and some developmental frames would be changed.**

Anarchical elements in free software development

Technologies (before marketing) have a tendency to take on functional names, indicating their specific technical character. So does free software. It is free software. „Free“ here means the entailment of a few degrees of freedom for its users. The specific claims have been outlined by Richard Stallman, the inventor of GNU, more than twenty years ago, as a reaction to some restrictive tendencies in software research and development. They are as follows: a user of software should be free to use a program for any purpose, to study its functions and fit it to own purposes, to make copies and propagate them to help others, to alter and develop the program and freely publish the results to promote the community and – resulting logically – he should be able to access the source code of the program[1]. Thus, free software can be developed by anyone who aquires the program which mostly (but not necessarily) includes that the program is freely downloadable somewhere.

Since this mode of software production has been introduced, it has had its own history. It took long years to actually develop and establish the first free operating system, GNU/ Linux, but ever since that has been achieved, free software has been flourishing. By now, free browsers, office and media applications and many other useful to funny free programs made their way onto ever more harddrives, much to the annoyment of their main commercial opponent: Bill Gates with his (sometimes) operating system Microsoft Windows.

But now what appears to be anarchical about free software? This association actually does not seem to appear with the initial programmers. Stallman and Torvalds for instance were mainly interested in securing free and open research conditions and in the technical task at hand. This was political in a rather detailed way, but apart from that, any far reaching political or even revolutionary implications directly referring to anarchism cannot be found. The association seems to mainly have been established by opponents and commentators from the press in the usual daily warmongering. They used the term „anarchy“ in its rather undefined colloquial meaning[2] to describe the specificly new phenomena of free programming. In this colloquial meaning, anarchy broadly describes a state in which no property exists, nor do rules or authorities and which thus has no stratified order in any (common) sense (which in addition is generally thought to result in nothing but sex and violence and the end of humankind within a couple of (sexy and violent) hours). Free software development now is associable with this state when it is seen as a kind of technological development and thus compared with the standard industrialized pattern for that. That pattern is normally strictly hierarchical, guided rigidly by corporate interest and controlled and organized through and through,

1 See the official definition under http://www.gnu.org/licenses/gpl.html.

2 This colloquial meaning can nowadays be found in many dictionaries. Anarchy there often equals orderlessness, total chaos and burning streets.

in every little step. This is of course owed to the huge amounts of money involved in any kind of industrial development and this also includes that it is largely a secret thing. The end result in a way gets published since it has to be sold, but the intermediary steps, specifically invented methods and all that is the companies property and not to be seen. Compared to this rigid framing of standardized industrial development, the developmental method of free software seems best described as: anarchy.
Of course this association initially also entailed some rhetorics. It also intended to draw on the sex and violence and end of anything image of anarchy. Gates for instance still tries to hold that free software development actually hinders a safe and high-quality development due to its lack of financial interest and directive order, thus posing a lethal threat to „good“ development in general. But in what followed, the association actually took on a rather positive value, mainly owed to two phenomena. First, there was the success. At least after Linus Torvalds' breakthrough with Linux, free software and its anarchical method turned from an exotic dream into a groundbreaking idea. Its development was fast compared to its dinosaur industrial rivals – and almost entirely without costs. Second, many scientists and programmers actually liked and embraced the anarchist image. Quite a few of them already were anarchists of course, especially the hackers and human rights activists, and did not feel uncomfortable with the term after all. But also the politically less exited gladly adopted the term to express a principle opposition to industrial methods: commercial programming and, as Stallman did, the propagating rule of ever more secret, „everyone for himself“-development. Both of these evils, now stretching out the dimensions of the dark side of the force in programming, have their rich and reckless emperor in the figure of Bill Gates and not at last due to his active engagement against free software, defining oneself as a (somewhat techno-)anarchist soon started to be a personal thing as well.
Radiating from a rather rhetorical reference to the free software production method as anarchical, we now find the establishment of an overarching general political attitude towards free programming as political action in a more general sense than merely within the copyright-debate, in turn placing the latter into its larger social context with capitalism. A crucial point about this identification is that the rhetorical use of the term „anarchy“ draws on the rather unfounded, preconceptive and largely false public image of anarchism briefly mentioned above. Thus the question arises, how free software actually compares to real anarchism as it is outlined by its own theories. It can be answered from the anarchist theory of technology.

The anarchist theory of technology

To state that something like an anarchist theory of technology exists might sound a bit strange at first hand. What could a political theory of technology be? How can technologies be political at all? They seem to be rather neutral, mere means to a great variety of ends. This is a very common view about technologies. But it is mistaken, even more so for the industrial age. We need to look at this mistake and will thus dwell a bit on the relation between technology and social order.
The relation arises from the fundamental insight which Marx had about the connection between production and social order. He stated that certain modes of production in the turn of history produce certain social orders. One very basic example for this is the division of work. It arose out of the knowledge how to grow crops and herd cattle which allowed prehistoric societies to gain excess production to store. This freed some of its members from the immediate need to produce food all the time and thus specialists could develop and the societies grew more complex and developed further. A political component got to this as some of the specialists became leader-specialists, priest-kings and the like. Thus a mode of production established a social order. This has now been the case ever since and at the core of every fundamental change in social orders, we can recognize some equally fundamental change in the modes of production.
Many of these changes have actually been technological changes. The above example already suggests this since the knowledge on growing and herding is largely technical knowledge. Thus the relation between production and social order has a significant technological component to it and it is

in this sense that technologies have to be deemed political. Technologies open up a specific range of possibilities for social orders. In most cases, this enlarges the range of human actions, but since the industrial revolution, technology also had a limiting notion to it. The complex tool-compounds called „machines“, by means of their design, their size and complexity, have to carry a number of necessities like conditions on how to handle them, a special division of work, sometimes specific hierarchies and so on. Thus the specific range of social possibilities opened up by technologies is not only a positive thing, free to choose. With the advent of machines, technologies also entailed social necessities. Marx had recognized this and within his concept to actually turn the tables by not having production dictate social order but social order dictate production (in communism), the appropriate technological change to allow this played a significant part as the „scientific-technological revolution“ which has been a constant issue to all communist societies ever since[3].

What remains to note from these considerations is the fact that technologies open up specific possibilites for social change and this is a point of consideration for anarchism much like it has been for communism. We will first have to ask ourselves which technologies would be needed in an anarchical society and this will point to some general technological characteristics fundamental to positively promote anarchism as a fundamentally new social order. Following these, we will additionally be able to state a few negative demands as well, more or less as the negations of the positive characteristics, to not only state what should be the technological case, but also what it should definitely not be. The positive characteristics will then enable us to recognize genuine anarchical technological structures whereas the negative demands will point us to structures which oppose anarchism.

Now let us sketch the anarchist theory of technology.

Like Marx, we will intend to turn the tables and have our social order dictate the mode of production, not vice versa. The basis of our thoughts should thus be the positive anarchist vision of the free, decentralized anarchist community. Which mode of production does it need? This is easy: a decentral, local production. Only then will people be able to live autonomously and thus free from outside rulers as we noted above. But how can this be achieved? Our current, globalized economy is quite the opposite of this. It grows things in one country, boils them in another, packs them in yet another and finally sells them somewhere entirely else. Can this irrational organization be rearranged to local models? We believe: yes. And as we mentioned above, this is to a large extent a technological task. Here, we meet Murray Bookchin. He is a known socialist and anarchist and he has written about the technological foundations of anarchism[4]. His thoughts are quite logical. We need to develop localized technologies which are able to gather ressources and produce goods in the most easy and comforable fashion possible. They should be workable by only a few people (the less the better, but up to a hundred probably if we imagine standard sized communities of 1000 to 2000 individuals), they should be able to regulate themselves and even repair themselves if possible[5]. This somewhat points to the old Enlightenment conception of technology[6] which has also been at the core of the communist scientific-technological revolution: technology as the saviour which abolishes all work for humans so they are free to live the lifes they want to live without being bound to the ugly necessities of daily production for daily survival. As such, anarchism is substantially technology-friendly, even very dependent on it. Entirely without it, work would much likelier have to be regulated, thus administered again. In addition, anarchism even embraces the highest possible state of technology, that of a full automation, to fully liberate humankind – at least with Bookchin.

3 It is a very interesting thing to view the history of the cold war, the collapse of the Soviet Union and the current development of high-tech as at least partly a result of these different technological cultures. David Hambling has written a very good book on technology transfer from the military to the civilian sector which sketches this situation: David Hambling: *Weapons Grade;* London 2005

4 For instance in: Murray Bookchin: *Für eine befreiende Technologie;* in: Hans Peter Duerr (Hg.): *Unter dem Pflaster liegt der Strand – Anarchismus heute;* Bd. 2; Berlin 1980

5 Science Fiction has a number of versions here. One of my favorites is: Herbert Franke: *Einsteins Erben;* Frankfurt am Main 1980

6 A central utopian vision of that can be found in: Francis Bacon: *Novum Atlantis;* Davidson 1989

And in that case, Bookchin holds, computers play a significant part as well. They will eventually conceptualize the work, steer the machines, administer everything and so on. Within his further theory, Bookchin also states that we have to achieve an equilibrium between humans and nature. This is important so the ressources are not wasted too excessively such that they become exhausted. A lack of ressources has always been a good reason for a war, so that should be avoided. But this does not have to be explored any further.
For now, this brief look will suffice to demonstrate the technological task at hand for anarchism. We can now state the basic positive technological characteristics anarchism needs. Its technologies will have to be such that they can be produced and maintained locally and they have to enable small communities to freely, easily and with the least possible amount of work produce their commodities. Further, they should be such, that they do not irrevocably exhaust local ressources or need very exotic ones.
Technologies which comply with these characteristics would not necessarily need a hiercharical social order any more. They would not create and reinforce dependencies from owners of ressources or lacks of ressources, from central monopolizing producers, highly skilled specialists and so on. Thus they would have a liberating effect on a society and could be called anarchical in a very close and genuine sense. Bookchin also mentions an example of such a genuine anarchical technology: the sun-furnace. It uses photovoltaic cells to produce more than 5000 degrees celsius, it can thus melt iron and steel, it's easy to build and maintain and can be operated by just a handful of unskilled workers. Thus it promotes anarchical interests in a genuine, clear-cut way. It promotes freedom and equalness and diminishes dependence and asymmetric inequalities as they are transparent in current industrial production.
This directly leads us to our negative demands. After we have seen which technological characteristics are demanded by a free humankind, we can also state which kinds of technologies hinder the development of a free society. Such technologies are namely designed in such a way that they need or reproduce the principles of authority and hierarchy or that of exchange-value, either in their production or in their later use. A clear example for this is current steel production. Steelworks as machine-compounds often need thousands of workers, from iron ore mining to melting to transportation and administration. Thus they rather propose hierarchical structures. They need specialists, a central organization, authority. And they suggest larger communities. The workers will need food, housing and entertainment and that leads to larger cities as we find them frequently with steel production. All this opposes the anarchical ideas of non-hierarchical structures, autonomy, decentralized, smaller communities, of comparatively free choices and little work. Thus we will demand of technologies to be free of such wanting structures which invite social complexity and class-construction, thus suppression and hierarchy.
Of course, the negative somewhat follows from the positive. But it still felt necessary to me to mention it to achieve a contrast between what we can now call a genuine anarchical technology (as the sun-furnace), a technology somewhat opposed to anarchism (as the huge steelworks) and probably technologies which could be deemed neutral from an anarchist point of view (as a simple excavator probably)[7].

Free software as a technology

Now let us look at free software as a technology. Is it genuine anarchical, opposed to anarchism or rather neutral?
We will approach this question by first stating some clear intuitions. First we have to note, that software is a technology basically in charge of controlling other things. It steers machines, tells them what to do. As such, it is not a technology on its own. It is always combined with some other machine which it controls. In the case of free software, this is always a common personal computer,

7 Although some of my colleagues might hold that one cannot sensibly speak of any neutral technology. See for instance: Christoph Hubig: *Technik- und Wissenschaftsethik – ein Leitfaden;* Leipzig 1993

not a harvesting machine, a steel press or anything[8]. So without personal computers, there is little use for free software but playing frisbee with the CDs. Thus we have to ask ourselves, in which way a personal computer can be seen as an anarchical technology if it is being controlled by freely developed software. To repeat: we can state that such computers could be rated genuine anarchical if they would promote decentralization, autonomy and the creative and free development of humans. If on the other hand such computers are rather proliferating control, the concentration of capital, centralization and such, they should be viewed as rather opposed to anarchism – in this case: whatever the mode of production of its software was.

The verdict is reached quickly. It stems from the current production of personal computers. This production is highly monopolized, in very centralized structures and the assembly-lines are globalized and exploit the poverty of foreign countries by means of financial power. This holds for any major personal computer brand. Personal computers have to be viewed as a typical product of a high industrialization, involving all of its clearly anti-anarchical structures[9]. Thus free software as a technology is clearly opposed to anarchism by means of its current dependence on personal computers. The included obligation to buy a personal computer affirms and reinforces the corresponding industry and the principles of its conduct.

A tough verdict which cannot be different – based on the current version of PC-production. We can now note that in addition to the concept of free software, a concept of free hardware (so to speak) would be needed as well to render free software into an anarchical technology. We might question though whether such a thing is even conceivable. The production of up-to-date PCs is so highly specialized, drawing on extremly specific components and ressources, that it seems rather impossible to conceptualize any decentral, small and local fabrication for them. To achieve this, much more research and development would first have to be done into a direction entirely different than the current, industrial one. Thus, a concept of „free hardware“ for the bearers of the current versions of free software, the up-to-date PCs, seems still a little too ficitious to help the concept of free software onto an at least principly safe ground. Here, Bookchin has to be critized, or at least relativized as well. He placed great emphasis on computers for the liberation of technologies, but that was back in the 60ies and the computer Bookchin mentions as an example is the DDP-124 from the Computer Control Company in Framingham. This was still a rather simple device (although it already included ICs), not too demanding in its production and ressources and he probably didn't foresee the development computers would subsequently undergo.

However, in comparison to these negative judgements, a point can be made about a sensibly anarchical free software as well. It could be conceivable, given our current technological situation, to program free software for simple computers which are in use in simple machines as the ones I have mentioned above: harvester, steel presses or things like that. Given the case that these simple machines and computers can principly be constructed and handled locally and decentral, without an immediate dependence on large-scale industries, free software would very genuinly promote a technological liberation. To give a case in which this would be of actual relevance, Siemens and its „engagement“ in third world development can be stated. Siemens has good contracts with development agencies and institutes and in turn provides the third world with a lot of machines for agriculture, water and waste management and so on. The machines are relatively simple to handle, but they draw on specific parts and specialized machine-programming, thus securing Siemens a huge and very dependent aftermarket. Free machine-software in this case for the Siemens-machines

8 At least I don't know of any other case. Most machine software is yet too specialized and too close to its producing industries.

9 Another thing (though not of direct importance) is the bounds these corporations share with the arms industry which provides the material basis for any exploitation and suppression. Just to state some historically popular examples: Wallace Eckert produced the IBM Model 701 in 1952 to calculate nautical almanachs for the US Navy's submarines. The 701 became the predecessor of all personal computers. Robert Noyce, the founder of Intel, conceptualized the first IC in 1959 which was further developed within Texas Instruments for the Solid Circuit Network Computer, a miniature computer to help the USAF with missile guidance in the Minuteman program (- not for the space program as is sometimes claimed). Current companies of course are still involved in these businesses.

would greatly proliferate the freedom of the concerned countries. However, the free software movement is not found here.
Thus the political difference free software as a technology makes is rather small, not to say: zero. In its current shape, it is only operated in personal computers, but these are severly monopolized and reinforce centralization. Buying computers promotes the according industries with affirmation and financial support. In addition, personal computers have developed into a highly advanced and technically demanding state which fundamentally hinders even the conception of a thing like „free hardware“. Despite this, there could still be a genuine free software, namely one which would be developed especially for more genuine anarchical machines, respectively the (accordingly rather simple) machine-computers which run them. This would be a more directly political action, in support of technological freedom. But here, free software doesn't seem to exist, at least not as a part of the free software movement. Resulting from these consideration, free software as a technology has to be regarded as rather opposed to the idea of anarchism. The term doesn't fit here at all.

Free software as a method
But the case is not closed yet. After our investigation into the implications of free software as a technology, we can proceed to another consideration: what about the production method of free software development as a pure method? Can it be regarded as anarchical? Does it promote or does it restrict autonomy and freedom? The case is not quite unambigious. It does of course operate without much apparent hierarchy or authority, it is decentralized and free for all (who already own a computer) and these things are good and have an anarchical potential in a different aspect, apart from the method towards which we will return in the next chapter. But on the other hand, we also find restrictive tendencies. Not only do still too little free operating systems exist to speak of a truly decentralized, non-authoritative method. Linux is still a very central frame here. Free software development also uses a lot of the capitalist vocabulary, linguistic institutions and rules as a basis for its daily business. This starts with GPL and the five rules for free software development and currently ends with the Creative Commons Licenses. Sure: these rules are intended to guarantee freedom. But at this point, Foucault can be mentioned[10], who has quite rightly wondered about where humans got this nonsense idea from that freedom has to be guaranteed by rules and institutions. Freedom, by definition, is just the very absence of rules and institutions and never has either of one really proliferated any real freedom. Rules and institutions, even as moderate guidelines, are restrictive, hierarchical and authoritative by nature, they cannot reasonably be associated with freedom. Here, free software development as a method fails significantly in providing a genuine anarchical framework for any subsequent work.
In company of these regulations, we often find another peculiar phenomenon which also weakens the alleged anarchical character of free software in its non-capitalistic aspect: free software is often being measured and valued in terms of its final proliferation of capitalism. Quite a few promoters actually argue that the developmental method is „effective“ not because it proliferates freedom and autonomy, but because it has a tremendous developmental output with low costs which can in turn be used to promote businesses. These lines of arguments then continue to state the many new businesses which have opened up on the back of Linux and how many big computer companies actually already profit significantly from free software. If the developmental method of free software is being measured in this way by its own promoters, its intentions in as far as what these finally aim at cannot appear anarchical in any *genuine* way anymore which in turn renders the method at least politically questionable.
In sum, free software development as a method appears counter-capitalistic or anarchical only in a somewhat short-sighted and premature manner, despite its own understanding. As so many counter-movements do (if it can be accredited with this much of a meaning beyond press rhetorics at all), it uses the very methods and words of its opponents in their negation to define itself as different. And

10 From the first chapter of: Jürgen Mümken: *Freiheit, Individualität und Subjektivität;* Frankfurt am Main 2003

by doing so, it doesn't recognize that it still operates within capitalism and authority, just by the very use of its words and methods, even if they are in negation. To use one of our earlier insights on this: free software still shares and reproduces the false beliefs in capitalism and hierarchy because it defines itself against it which also entails to accept its existence and use its methods and standards.
It is in this sense that we have to judge the method of free software as bracketed by the ideological frameworks of capitalism and hierarchy. The extent to which this misbelief is actually embedded in free software can be shown in Toby Milsom who stated that the GPL uses copyright to express anarchism. A strange idea.

The overall rating of free software development doesn't look good. It only happens within industrialized computers, thus it doesn't help us build a decentralized society but rather promotes the opposite. And even as a pure method, it still reproduces capitalism and authority by using capitalist methods, concepts and standards. To judge it anything anarchical or somewhat revolutionarily political seems a rather strange idea. Thus the connection between anarchism and free software in the current debates and given the current situation of free software is really nothing but rhetorics. Its invention has to be accredited to the false public image of anarchism, the colloquial understanding of the term and its unreflected use.

Free software as a demonstration of successful anarchy

But is this really the end of it? Does free software development really not have anything political to it? Nothing anarchical? Well: there could still be a few things. We already mentioned one: free software for less dependent machines. But another, a demonstrative value can be obtained from free software development as well. And this is a notion which we will develop now out of the few anarchical *aspects* which can be found.
To do this, we will first have to turn to an argument within which the idea of anarchism in free software development played its significant role, namely the productivity-argument for free software development, directed against intellectual property rights and software patents[11]. It is a very central argument in the whole story, but until now, we were able to operate without it since we were rather concerned with the general relation between free software development and anarchism in a political sense, its factual appearance and any possible shapes. The productivity-argument however is more of an argument about what is good for software development, not about what is good for anarchism – which is why we postponed it. At this point however, we will focus on this argument and show how it can be reinterpreted to state something essential about anarchism.
First to state the argument: the productivity-argument holds that the anarchical mode of production, in so far as it is unguided, open for everyone and not profit-oriented, yields a very good and sophisticated development, in fact, it even appears to be better than its capitalist counterpart in many possible comparisons[12]. This is easily empirically proven and it is intended as a counter-argument to the capitalists idea that real development needs profit as a stimulus and a structure to work efficiently and it thus has been the core reason against the extension of intellectual property rights.
Now in its current shape, this is an argument which basically states that a partly anarchical mode of production is actually good for software development. And we were able to show that the notion of anarchism which is used in this argument is not to be outrightly equated with anarchism in its full theoretical sense, but rather in a colloquial meaning. But since the colloquial meaning is not entirely far out, only partially, the argument still has an attractive potential for anarchist theory, if we reformulate it a little into the following: an anarchical mode of production is more productive and yields a better development than the capitalistic mode of production.

11 See for instance: Lawrence Lessig and Richard Epstein: *Geistiges Eigentum;* Technology Review 7/ 2005 (German ed.)

12 Philosophy and sociology of science have shown us of course that comparisons can prove a lot, especially if such large scale, mulit-factored phenomena have to accredited. But some of the more reasonable ones support the claim.

This is a legitimite reformulation (in fact, a simplification) of the productivity-argument and now, the focus is on anarchism and software development can be taken to be a mere example. In this case, the validity of the argument, which would substantially draw on free software development as a case study, suggests that our current authoritative-capitalist order of the world not only has those many evils which it has already been accused of so frequently. In addition, it would have proven that the authoritative-capitalist mode of production is not even effective as such. Thus seen from a global point of view, it is irrational to maintain the authoritative-capitalist world order not only in light of all its negative side-effects, but also as a mode of generally maintaining and promoting humankind. Because it does neither as good as an anarchical order would do. In comparison, it actually even hinders the development of humankind. Such a conclusion would be of great importance since this topic so far has been only a matter of intellectual debate. Free software now could help this debate onto an empirical footing, it could state a case in point for anarchism.
However, we will still have to accredit our critical remarks on behalf of free software production as a method. The mere existence of the few anarchical characterstics which initially invited the equation does not suffice. To make the case of free software development a good and truly valid example, we have to stratify it. Free software would have to have been produced entirely without any allegiance to rules, authorities or licenses. Any development restricted by such regulative ideas can not be regarded as a genuine anarchical development. Only if better development also takes place in their total absence, free software development can be accepted as an example for a better, more productive and more creative humankind in absence of an authoritative-capitalist order.

Conclusion: if free software is developed in a genuine anarchist fashion, it gains a significant political role as well. It's still only in a sandbox, but a sandbox with a substantial and general demonstrative value.

References

- Antonio Negri/ Michael Hardt: *Die Arbeit des Dionysos: materialistische Staatskritik in der Postmoderne*. Berlin [u.a.]: Ed. ID-Archiv, 1997
- Ulrich Enzensberger: *Die Jahre der Kommune I: Berlin 1967 – 1969;* Köln 2004
- Karl Marx: *Das Kapital*; Stuttgart 1957 (Kautsky-Translation for Kröner)
- Clifford Harper in his *Anarchy: A Graphic Guide*, Hampden Press 1987
- Peter Heintz: *Anarchismus und Gegenwart*; Berlin 1985
- Gilles Deleuze/ Félix Guattari: *Anti-Ödipus;* Frankfurt am Main 1995
- Michael Hardt/ Antonio Negri: *Empire;* Cambridge (Mass). 2000
- Herbert Marcuse: *One Dimensional Man;* 2.ed, London 2002
- Joan Nordquist (ed): *Anarchism: contemporary theories: a bibliography;* Santa Cruz (Calif.) 1999
- David Hambling: *Weapons Grade;* London 2005
- Murray Bookchin: *Für eine befreiende Technologie;* in: Hans Peter Duerr (Hg.): *Unter dem Pflaster liegt der Strand – Anarchismus heute;* Bd. 2; Berlin 1980
- Herbert Franke: *Einsteins Erben;* Frankfurt am Main 1980
- Francis Bacon: *Novum Atlantis*
- Christoph Hubig: *Technik- und Wissenschaftsethik – ein Leitfaden;* Leipzig 1993
- Siva Vaidhyanathan: *The Anarchist in the Library;* New York 2004
- Jürgen Mümken: *Freiheit, Individualität und Subjektivität;* Frankfurt am Main 2003
- Lawrence Lessig and Richard Epstein: *Geistiges Eigentum;* Technology Review 7/ 2005 (German ed.)

And a guide to anarchism can be found under: http://sourceforge.net/projects/anarchism.