Images of innovation in discourses of free and open source software by George Dafermos and Michel J.G. van Eeten

Abstract

In this study, we examine the relationship between innovation and free/open source software (FOSS) based on the views of contributors to FOSS projects, using Q methodology as a method of discourse analysis to make visible the positions held by FOSS contributors and identify the discourses encountered in the FOSS community. In specific, our analysis reveals four discourses: four ways of expressing oneself used by FOSS contributors, which, aside from certain commonalities, postulate fundamentally different conceptions of innovation. Whereas the dispersion of FOSS contributors' subjectivity across four different discourses is indicative of the diversity and heterogeneity of the FOSS community, their commonalities, however, demarcate a common ground that all discourses share: points of agreement include the negative effect of patents on innovation, the predominant role of end users over manufacturers in the innovation process and the embrace of FOSS licenses as a key enabler of innovation. In the conclusion, we outline some implications for innovation management and policy.

Contents

[Introduction](#)
[Methodology](#)
[Results: Unfolding the discourses](#)
[Implications for innovation policy and management](#)

Introduction

Research interest in software developed and distributed freely over the Internet by voluntary associations of hackers [1] known as free and open source software (FOSS) projects has been steadily increasing during the last ten years. Readers of *First Monday (FM)* are familiar with the topic. In fact, few journals have devoted as much attention to FOSS as *FM*, as can be seen from the publication of Eric Raymond's (1998) 'The cathedral and the bazaar,' which popularised the study of FOSS, and the October 2005 *Special issue on open source* (<http://firstmonday.org/issue/view/212>), to name but two milestones in research on the topic. Since, a plethora of works published in *FM* have highlighted the uniqueness of FOSS. Characteristically, Schweik and English (2007) remarked the striking divergence of the mode of governance of FOSS from that typically encountered in natural resource commons on account of the informal and non-hierarchical governance of the former, while Jordan (2009) emphasised the 'novel understanding of property' encapsulated in the development of FOSS, which effectively inverts 'property as exclusion to property as distribution'. The ever-increasing stream of contributions to the evolving FOSS literature attests to the fact that the scholarly attention devoted to FOSS has since only been increasing, often in directions that extend beyond the realm of software production. Indicatively, Scacchi (2010) concluded that FOSS projects 'are in many ways socio-technical experiments to prototype alternative visions of what innovative systems might be in the near future' on the basis of a mode of governance that effects a 'transformation in the marketplace of ideas and [in] the means of production from centralized authority with corporate enterprises, towards a more decentralized commons-based peer production'. On the same wavelength, Stevenson (2010) called attention to FOSS as an 'organizing vehicle for the world's information workers' and as 'a means to create more open and democratic political systems' (see also Stevenson, 2007), while McNally, *et al.* (2012), in a more recent paper that traces the policy implications of FOSS (and user-generated content, more broadly), underlined the way in which FOSS 'generates value, serves as a medium for cultural expression and allows innovative activity' (see also McKenzie, *et al.*, 2012). Common to all these works is that FOSS is depicted as innovative.

Upon closer scrutiny, however, the literature reveals a striking divergence of views on the relationship of innovation and FOSS. Characteristic is the discussion in the literature concerning the group size that best conduces to a climate of innovation. According to Jamie Zawinski (1999), former leader of the Mozilla Project,

great things are accomplished by small groups of people who are *driven*, who have unity of purpose. The more people involved, the slower and stupider their union is.

Zawinski is not alone in holding that large development groups inhibit innovation. Similar views have been expressed by other practitioners (*e.g.*, Eich and Hyatt, 2003). Others though have been led to radically different conclusions. Valloppillil's (1998) analysis of the FOSS development model underlines the potential for innovation unleashed by opening up participation to a multitude of programmers scattered the world over:

The ability of the OSS [open source software] process to collect and harness the collective IQ of thousands of individuals across the Internet is simply amazing.

To illustrate the 'tension' in the literature with one more example, let us look at the discourses touching upon the issue of FOSS licensing. At first sight, the discussion appears polarised between two contending arguments. On one side, it is claimed that FOSS licensing structures foster innovation (*e.g.*, Benkler, 2006; Franck and Jungwirth, 2003; von Hippel and von Krogh, 2006). This view is epitomised in full swing in Osterloh and Rota's (2007) tribe on FOSS:

OSS [open source software] licenses are an ingenious institutional innovation that mark the main difference to the other cases of collective invention. They are the root of a new innovation model, which goes beyond traditional forms of markets, hierarchies and contractual strategic alliances. [2]

This view on the effect of FOSS licenses on innovation, however, is not commonly accepted. Its critics counter-argue that 'the GNU GPL [3] [and FOSS licensing, in general] stifles innovation' (Weston, 2004). Indicatively, to substantiate this point, Passell (2003) refers to the case of research in medicine, contending that

If federal research in medicine had been distributed under some equivalent of the [GNU] GPL, the spectacular burst of innovation in drugs and genetic engineering by private enterprise in the last decade would have been delayed.

Clearly, the latter statement is at variance with the former one. Most important, this tension is not confined to the question of group size or software licensing alone but rather spans the entire spectrum of FOSS discourses. The fact that actors occupy diametrically opposite positions indicates the participation of FOSS discourses in a discursive struggle over the meaning and significance of the FOSS phenomenon. From the perspective of Foucauldian discourse analysis, which views discourses not as representations of the world but as practices acting upon it (Foucault, 2002), FOSS is a field of contestation among occupants of different positions in which the character of FOSS development is bound up with questions of identity and ethics (Iivari, 2010). In consequence, elucidating the positions held by actors in the FOSS community calls for a more elaborate approach toward the analysis of the discourses that are being articulated across the field of FOSS development.

The remainder of this paper is structured as follows: in the [next section](#) we describe the research methodology and data sources, clarifying our discourse analytic approach based on the employment of the Q methodology. The empirical findings — four distinct FOSS discourses — are presented in the [follow-up section](#). The [final section](#) concludes by sketching out the implications for innovation policy and management.

Methodology

To make visible the subject positions held by contributors to FOSS projects, rather than those of analysts or theorists, we employ the Q methodology (Brown, 1980; Stephenson, 1953), a method of discourse analysis particularly 'suited to the task of uncovering positions really held by participants in a debate rather than accepting decision-makers', analysts', or even the participants' predefined categories.' [4]

The implementation of a Q study begins with the construction of the *Concourse* [5] wherein all statements relevant to the subject are collected. In our case, the organising principle for its construction was the occurrence of the word *innovation* in FOSS discourses [6]. The *concourse* initially comprised about 200 statements but after the elimination of largely overlapping statements the number was reduced to 57 in the final version, called *Q-set* [7].

With the Q-set ready for administration, we proceeded to find an appropriate group of persons to perform the main stage of the procedure. This group, called *P-set*, is not constructed through random selection but in accordance with the principles of discourse analysis — '*first question: who is speaking?*' [8] — being therefore 'a structured sample of ... persons who are expected to have a clear and distinct viewpoint regarding the problem' [9]. As a first step in that process, we contacted six individuals who are known FOSS experts in the Netherlands (purely out of practical considerations, given our whereabouts) and asked them to recommend others that in their opinion we should contact in order to perform the main stage of the procedure. This method (known as snowball sampling [*e.g.*, Biernacki and Waldorf, 1981]) provided us with a system of referrals that we used to construct the P-set, resulting in the selection of 37 individuals whose background reflects the diversity of the FOSS community, encompassing programmers, advocates and authors of documentation, as do their project affiliations, encompassing both large and small projects. What is important in this sampling approach is not so much the number of participants but their *diversity* so that whatever subjective factors (*i.e.*, points of view) exist will have ample opportunity to reveal themselves. The aim is to show the ways in which the FOSS community is subjectively segmented: the nature of the different ways to view the relationship between FOSS and innovation. Therefore, 'since the interest of Q methodology is in the nature of the segments and the extent to which they are similar or dissimilar ... the issue of large numbers, so fundamental to most social research, is rendered relatively unimportant' [10]. The fact, however, that the P-set is geographically confined (for practical reasons, as aforementioned) to the Netherlands, where 31 of the 37 study respondents are located [11], may limit the generalisability of our findings across different countries and beyond the European continent.

All 37 respondents were asked to perform a *Q-sort*, which is the main stage of the procedure. Each Q-sort was operationalised with a set of randomly numbered cards, each card being a different statement, which the respondents had to rank-order (sort) according to the degree they agreed or disagreed along a continuum from 1 for 'most disagree' to 7 for 'most agree', with 4 denoting indifference. To ensure that no relevant statements were missing, we asked six more persons to perform a preliminary Q-sort and to identify any missing statements. In this way six more statements were added to the Q-set [12]. In standard Q-sort fashion, the respondents were asked to place the statements in a quasi-

normal distribution, forcing them to prioritise their choices. In addition, respondents were asked to elaborate on their placements. Upon the completion of the Q-sort, they were asked to name two FOSS innovations, justifying their choices; and to recommend others for a Q-sort based on their having a unique expert viewpoint. The first question aimed at complementing the statistical analysis with a 'qualitative picture' that might prove useful in allowing for a richer interpretation of findings. The second question provided us with a system of referrals whereby we contacted and selected persons to participate in the research.

Following their collection, these rankings were superimposed in a correlation-matrix and factor-analysed: the rankings, in the form of the correlation matrix of all Q-sorts, were subjected to centroid factor-analysis and rotated according to the varimax principle in order to illuminate 'the degree of (dis)similarity in the points of view' [13]. In this way, four factors were derived from the analysis. Although Q methodology is a factor-analytic procedure, it reverses conventional factor analysis, as 'it correlates persons instead of tests ... and is used to describe a population of viewpoints and not, like in R, a population of people' (van Exel and de Graaf, 2005). The resulting factors are essentially crystallisations of *discourses*, meaning certain ways of speaking that 'systematically form the objects of which they speak' [14]; 'a group of statements which provide a language for talking about ... a particular topic at a particular historical moment' [15].

<i>Number of statements</i>	3	6	14	17	14	6	3
<i>Statement scores</i>	-3	-2	-1	0	1	2	3
	most						most
	disagree						agree

Figure 1: The opinion continuum for the Q-sort.

Table 1 shows the correlations between the respondents and the factors. The discourses are best understood by analysing how the statements which characterise the factor fit together. Particularly important are the statements at the extreme poles of the continuum (see **Figure 1**): most disagree (-3) and most agree (3). Based on these findings, a description of the discourses is given in the [next section](#).

Table 1: Correlation between respondents and factors.					
<i>Respondent</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	
1	0.5258X	0.0793	0.2977	0.2771	
2	0.5958X	0.1739	0.3583	0.3729	
3	0.5148	-0.1261	0.4163	0.2805	
4	0.5712X	0.2717	0.1236	0.1113	
5	0.1686	0.3493	0.5306X	0.0655	
6	0.4345	0.0203	0.5396	0.3962	
7	0.5513X	0.3268	0.1768	0.3941	
8	0.2353	0.1354	0.5081X	0.3655	
9	0.6230X	0.1191	0.3026	0.2195	
10	0.1602	0.1480	0.2663	0.3432	
11	0.4784	0.2057	0.1371	0.5659X	
12	0.2850	0.0680	0.2107	0.5771X	
13	0.1823	0.2099	0.5165X	-0.0668	
14	0.1674	0.2410	0.0363	0.5758X	
15	0.3767	0.3321	0.3821	0.1942	
16	0.2115	0.2157	0.4817	0.5009	
17	0.2800	-0.0343	0.2895	0.4767X	
18	0.5001X	0.1760	0.2693	0.1733	
19	0.4162	0.1143	0.0466	0.5092X	
20	-0.0065	0.1053	0.0902	0.7604X	
21	0.1977	-0.1274	0.5403	0.6092X	
22	0.1720	-0.0187	0.5982X	0.4038	
23	0.5473X	0.1244	0.2092	0.3697	
24	0.4030	0.2511	0.0714	0.7660X	
25	0.6860X	0.1187	0.0946	0.2316	
26	0.1991	-0.0557	0.0315	0.4561X	
27	0.3663X	0.0634	0.2781	0.2001	
28	0.3997	0.4847	0.2985	0.1034	
29	0.2804	-0.0831	0.2506	0.5204X	
30	0.1035	-0.0613	0.1346	-0.0143	
31	0.1546	0.0825	0.3643	0.3994	
32	0.1084	0.4967	0.2284	0.5674X	
33	-0.0051	0.5947X	0.1118	0.0788	
34	0.2063	0.1969	0.5862X	0.4882	
35	0.4550	0.0876	0.4265	0.4323	
36	0.1255	0.3186X	-0.0801	-0.0223	
37	-0.2396	-0.5547X	-0.3215	-0.3721	
% expl.Var.	14	6	11	17	= 48

Results: Unfolding the discourses

Table 2 states the arguments that sum up each of the four discourses:

Table 2: Discourses on free and open source software development.	
Discourse A	Innovation is driven by users: the more end users become involved in the production process the greater the potential for radical innovation.
Discourse B	Innovation inheres in small groups and is facilitated by the free availability of elegant source code.
Discourse C	Innovation occurs at the interface between businesses and open source communities and is characterised by imitation.
Discourse D	Innovation is enabled by open standards, which lay the foundation for the emergence of open collaborative structures.

Commonalities

Prior to describing the dispersion of FOSS developers' subjectivity across these discourses, we must note the commonality of views manifest through statements that permeate all discourses.

Table 3

	A	B	C	D
The GNU GPL stifles innovation and impedes reuse. Exactly the kinds of things you ostensibly wouldn't want an "open source" license to do. [5]	-2	-1	-3	-3
At best, software patents have thus far had a neutral effect on software innovation; at worst, they have had a significant negative impact on future innovation. And there is simply no evidence that patents are needed to promote innovation in software. [6]	2	3	3	3
Innovation requires a level of risk, and the returns will never justify the risk when the playing field has been levelled by an open source philosophy. [7]	-1	-1	-2	-1
The hottest innovations will always come from well capitalized start-ups driven by the carrot of wealth. [47]	-3	-2	-2	-2
Without the sustained investment made possible by commercial software, real innovation would not be possible. [49]	-2	-1	-3	-2
"Functionally novel" innovations are significantly more likely to be developed by manufacturers rather than by users. [13]	-3	-2	-1	-1

All discourses view the GNU GPL as a key enabling condition of innovation, offering sufficient protection against the danger of private expropriation of the commons and giving a spur to wider participation by the surrounding community of users and developers in the innovation process. In contradistinction, patents are at best irrelevant and at worst harmful to innovation. In effect, they remove the incentive for adoption and further development of a technology by other technologists and thus render impossible the implementation of the open source development model.

Taken as a whole, the FOSS ecosystem proves that innovation is not a question of financial funding. Real innovation does not depend on the economic investment made possible by commercial software; it is independent of it. Arguably, given that end users are a more potent source of 'functionally novel' innovations than manufacturers, innovation would still thrive even if the innovation process were to be 'open-sourced' and decoupled from the market economy.

Table 4				
	A	B	C	D
Government should not develop innovation policies to fund open source communities as competitors to proprietary incumbents. [24]	-1	-1	-1	-3
Open source is a technology innovation strategy looking for a business model. [34]	-1	-3	-1	-1
If companies are allowed to tap the unpaid, innovative labour of the community, in-house and waged labour will be pushed out by the market imperative to cut down on personnel expenses. [52]	-2	-3	-1	-2

The FOSS community provides a public good, which the government could fund to obviate the danger of underinvestment in its further development and thus ensure its sustainable provision to the public at large. However, under no circumstances should one conjecture from this that the community of FOSS developers is anxious to find a business model or that it needs one. Neither should one come to the erroneous conclusion that the increasing involvement of software firms in FOSS development is putting programmers' jobs in jeopardy, as the industry needs programmers to maintain and customise FOSS.

Table 5				
	A	B	C	D
Bazaar tends to be more of a copy machine than an innovator. Innovation comes when a product starts as a Cathedral. [19]	-2	-1	-1	-1
This 'sharing, collaboration and openness' among engineers and scientists is actually the backbone of innovation — under the skin, so to speak, of the conventional business practices of firms. [43]	3	2	2	1
Standards inhibit innovation. That's not always a bad thing, because it allows software to work together, but at its heart, a standard's purpose is to stop you from creating you own way of doing something. [50]	-1	-2	-1	-3
Open standards and user freedom of choice allows innovation, it even promotes innovation. Users get to choose, which provides an environment of evolution and of survival of the fittest. [51]	3	1	1	3
Linus put into practice an innovative and unorthodox model of developing software: "release early and often". Frequent releases and updates (several times in a week) are typical throughout the entire development period of Linux. In this way, Linus kept the community constantly stimulated by the rapid growth of the project and provided an extraordinarily effective mechanism of psychologically rewarding his co-developers for their contributions that were implemented in the last version. [45]	2	1	1	1

Innovation requires openness and transparency; it is itself the product of a culture of openness and knowledge sharing among engineers and scientists. What sets these collaborative practices in motion is open standards: standards establish an architecture of participation which enables innovation to take place at the edges of the network. The openness of standards is not however the only enabling condition of innovation. Equally important is the practice of releasing early and often, for it opens up the process of innovation from the early stages of development, encouraging participation whilst providing an effective mechanism to psychologically reward contributors.

Discourse A: Radical user innovation

The thrust of the first discourse is clearly illustrated by the three statements that it strongly endorses:

Table 6				
	A	B	C	D
The idea of open source itself is an innovation. [33]	3	2	-1	-1
This 'sharing, collaboration and openness' among engineers and scientists is actually the backbone of innovation — under the skin, so to speak, of the conventional business practices of firms. [43]	3	2	2	1
Open standards and user freedom of choice allows innovation, it even promotes innovation. Users get to choose, which provides an environment of evolution and of survival of the fittest. [51]	3	1	1	3

Open source *is* innovation *per se*: distributing production requirements across the network and opening up the production process so that anyone can contribute is a radical innovation in itself. As S. Kugler (respondent #9) [16], contributor to the KDE project, puts it: innovation is 'the development model itself: interacting with your users and dealing with things in an open way ... giving feedback all the way, everywhere'.

Sharing is a core principle of science and a key enabler of innovation. Engineering practice would simply have been unthinkable without a collegial ethic of sharing. Defined by the principles of sharing, collaboration, and openness underlying it, FOSS development belongs to this long tradition. But considered in its actual dimensions, open source is an essentially novel way of doing things. Not only are users the main factor of production but the production process itself is geographically distributed and driven, so to speak, by the wisdom of crowds.

By making distributed development possible, standards foster the ethos of sharing, collaboration and openness underlying innovation. It is not hard to discern the importance of open standards as a platform upon which the possibility to experiment and innovate is encouraged. Innovation thrives on open standards, for open standards are instrumental in enabling innovation to occur at the edges of the network, in the periphery where the real intelligence and capacity for innovation resides. In a word, open standards decentralise the capability to innovate across the network. In much the same way that standards are required for mass production, open standards are required for the emergence of mass innovation, where the wisdom of crowds supplants the secluded innovator.

According to two additional statements which further elucidate the discourse:

Table 7				
	A	B	C	D
"Functionally novel" innovations are significantly more likely to be developed by manufacturers rather than by users. [13]	-3	-2	-1	-1

Bazaar tends to be more of a copy machine than an innovator. Innovation comes when a product starts as a Cathedral. [19]	-2	-1	-1	-1
--	----	----	----	----

Innovation is driven by end users: it is they who, by adapting technology to their needs, realise its full potential. The reason why end users are so crucial to innovation is clear, considering that innovation is seldom the work of individual geniuses. Rather, innovation is an emergent property of collective action. Thriving on the contributions of a multitude of individuals scattered all over the world, the self-organising 'bazaar' of FOSS development affirms the potentiality for innovation engendered by opening up product development, from its embryonic stage, to the widest possible range of contributions by users.

Equally important in delineating this discourse, bringing its general outline to the fore, are the following three statements which it strongly opposes:

Table 8				
	A	B	C	D
It is unclear how innovative and sustainable open source can ultimately be. [28]	-2	0	0	-1
One thing that open source is not generally known for is innovation. In general, the rate of innovation in the closed source world is far faster because the desire to stay ahead and stay in business has always been stronger than the desire to re-implement something that's already been done. [36]	-3	0	-1	0
The hottest innovations will always come from well capitalized start-ups driven by the carrot of wealth. [47]	-3	-2	-2	-2

There is no doubt that FOSS is sustainable and innovative: the Internet, a radical innovation in networking with nearly 40 years of history of continuous innovation, is inseparable from FOSS: the core Internet infrastructure, from BIND, the most commonly used DNS server to Apache, the dominant HTTP server, is a veritable mosaic of FOSS.

The continued development of FOSS shows that the contribution of the profit motive to innovation is but marginal. As Olga Drossou (respondent #23), Information Society NGO activist, noted, 'the greatest innovation of the last decades, the Internet, was not driven by the motive of wealth; financing is not a precondition of innovation'.

To recap, open source is a radical innovation both in terms of the development model and the technologies produced in this way. The innovation of the FOSS development model consists foremost in the central role that end users perform: it demonstrates that innovation is driven by end users and so its potential is proportionate with their degree of participation.

Discourse B: Small-group innovation

In contradistinction to the first discourse which posits the radical character of innovation on the strength of pervasive user participation, the second discourse maintains that innovation is the product of small groups.

Table 9				
	A	B	C	D
More eyes on code means that bugs can be more easily and promptly characterised; however, neither the characterisation of bugs nor their fixing counts as innovation. [16]	0	3	1	1
Expand and improve isn't exactly "innovation". [17]	0	1	-1	1
With the coordination mechanism provided by the Net, innovation becomes faster and cheaper. However, this model does not clearly enhance innovation itself: ideas remain the domain of the chief architect. [15]	-1	-1	0	-1
The nature of the "Bazaar" model is what gets in the way of innovation. The more people you put onto a team, the more conservative they'll become. Innovation usually requires risk taking that just isn't possible when you're trying to please a lot of different people. [18]	-1	1	-1	-2

The scope of mass participation is limited to the task of characterising and fixing bugs, which, though important, does not count for innovation: innovation does not consist in expanding and improving. Most important, innovation 'never starts with a large group: people should know each other ... to bring a crowd together is counter-productive' (Marc Sutter, respondent #33, contributor to the CDL3 and GNU AGFL projects). Though valuable as it may be, the distributed, bazaar model of development comes after the phase of innovation, which involves only a small tightly-knit group of people.

Table 10				
	A	B	C	D
On-going parallel release structures could generate more innovation and continuous improvement. [8]	1	-2	1	0
Linus Torvalds is a great leader. [56]	1	-3	0	-1
Code elegance is crucial to software innovation. [62]	1	3	0	-2

In contrast with the first discourse, the second discourse is critical of parallel development and release structures, contending that they translate into waste and run counter to consistency. Moreover, this discourse is critical of centralised authority, which point it illustrates with reference to Linux: the rule that modifications contributed to the project have to be filtered and approved by the project leader, Linus Torvalds, results in unnecessary bottlenecks, which could be obviated through the distribution of authority across the base of Linux developers. Software innovation also calls for code written elegantly: aesthetic considerations aside, elegant code facilitates understanding so that one knows where to modify or add things.

Table 11				
	A	B	C	D
The idea of open source itself is an innovation. [33]	3	2	-1	-1
At best, software patents have thus far had a neutral effect on software innovation; at worst, they have had a significant negative impact on future innovation. And there is simply no evidence that patents are needed to promote innovation in software. [6]	2	3	3	3

Open source *is* innovation: the general availability of source code ensures that a project is open to changes in its environment, thereby taking on an evolutionary character. What ensures this openness of source code is the GNU GPL, which effectively precludes one's taking from the commons without giving back. As Donna Metzlar (respondent #36), contributor to the Codeyard project, says: 'you have to provide the source code ... this is what keeps a project open to development, open to evolution, open to innovation'. Patents, by stark contrast, serve to limit the wider applicability of existing ideas and their combination, which is often a powerful source of innovation, and are thus counter-innovative.

Table 12				
	A	B	C	D
Open source is a technology innovation strategy looking for a business model. [34]	-1	-3	-1	-1
If companies are allowed to tap the unpaid, innovative labour of the community, in-house and waged labour will be pushed out by the market imperative to cut down on personnel expenses. [52]	-2	-3	-1	-2
Open source principles could potentially be applied to almost any intellectual endeavor, and may be a very important innovation toward harnessing global collaboration toward solving complex public policy and management problems. [42]	2	-1	0	3

Open source is neither a business model nor is it looking for one: to develop FOSS, says LaTeX hacker Marleen Overkamp (respondent #37), 'you do not need a business model ... and open source is not to find money'. From this, of course, one should deduce neither that FOSS destroys traditional business models nor that it eliminates the need for waged programmers. For example, Red Hat is a commercial company that employs people to provide Linux-related services, and 'so open source creates more jobs than it destroys' (Marleen Overkamp, respondent #37). However, the wider application of open source principles to complex public policy and management problems is constrained by the fact 'that you cannot apply open source to bureaucratic processes' where 'you need rules and norms' (Donna Metzlar, respondent #36).

In a nutshell, this discourse holds that radical innovation originates in the collaborative practices of close-knit groups, whose members, united in their vision, give concrete shape to

the idea that contains the sperm of innovation. And the idea of open source, that source code be freely shared is a radical innovation: for it gives shape to a socio-technical environment that is conducive to evolution. According to this discourse, the effect of mass participation remains, though important, peripheral to innovation. And insofar as software innovation requires such a shared vision and understanding, which inheres in small groups, code elegance is crucial.

Discourse C: Sustaining business innovation

In contrast with the first two discourses which invest the concept of innovation with a disruptive character, the third discourse views innovation as an incremental, sustaining process. However, what clearly distinguishes the third discourse is its emphasis upon the role of the firm.

Table 13				
	A	B	C	D
Firms combine in-house expertise with the inventive activity in the OSS community to create innovations. [9]	1	2	3	1
Although the open source method for producing software is an important organizational innovation, much of the software that is being produced under this method today is not innovative in the sense that VisiCalc (the first spreadsheet) or WordStar (one of the first word processors) was. Indeed, most open source projects under way are meant to imitate (or clone) existing for-profit software. [21]	-1	1	1	-1

Highlighting the role of the firm in the innovation process, the third discourse holds that innovation occurs at the interface between businesses and the community of FOSS projects. Further, it argues that the production of FOSS is process-oriented: FOSS projects focus on developing better tools to more efficiently collaborate and communicate and on re-implementing available technology.

Table 14				
	A	B	C	D
Quick imitation robs innovation of value. [11]	-1	-2	-3	0
Open source allows small enterprises to afford innovation. [38]	1	-1	2	1
Without the sustained investment made possible by commercial software, real innovation would not be possible. [49]	-2	-1	-3	-2

et this does not mean that imitation is useless. Quite the contrary, imitation enhances the value of an innovation and the open source development model is a great imitator, slashing costs and shortening development cycles. Doing the same thing better and cheaper is essential for business innovation and helps explain why the business world, and small companies in particular which do not really fit into the licensing schemes of corporate behemoths, could benefit from FOSS. In many ways, innovation in the software industry has always been the domain of FOSS: prior to the creation of a mass market for commercial software in the 1970s, it was common practice in the industry to freely distribute the source code; whereas nowadays, it is interesting to observe 'how open source forces innovation into closed source software' (A. Hemel, respondent #13, attached to the GPL Violations project): for instance, the continuous inflow of improvements into Mozilla Firefox is forcing Microsoft to update Internet Explorer.

Table 15				
	A	B	C	D
The GNU GPL stifles innovation and impedes reuse. Exactly the kinds of things you ostensibly wouldn't want an "open source" license to do. [5]	-2	-1	-3	-3
Copyleft is the key innovation in the governance structure of open source, bringing people with different motivations, investors and donors into a symbiotic relationship. [4]	1	-2	3	1
At best, software patents have thus far had a neutral effect on software innovation; at worst, they have had a significant negative impact on future innovation. And there is simply no evidence that patents are needed to promote innovation in software. [6]	2	3	3	3
Reverse engineering is mistakenly seen as important to innovation and competition in all industrial contexts. [12]	0	0	-2	-2

It is understandable that some big firms, accustomed to exact any price they wish for their products with no justification other than the absence of competition, 'see the GNU GPL as a threat because it threatens their way of selling' (V. Sessink, respondent #8, contributor to Dutch OpenOffice project). But it is hard to see how increased competition could stifle innovation. In fact, it is owing to the GNU GPL that all kinds of people with a multiplicity of interests are attracted to FOSS and can work together. By contrast, the effect of patents, by raising the barriers of entry against small firms, is to entrench monopolies. And monopolistic practices, tantamount to the elimination of competition, hinder innovation. It is mostly in order to counter-balance the negative effects of such institutional devices that reverse-engineering exists: 'reverse-engineering was introduced to balance between the rights of the author ... and the rights of the industrial customer to buy a product and still maintain independence from the software vendor. Without this independence, the rights of the software vendor would damage innovation and competition' (copyleft activist T. Vardaris, respondent #22).

Table 16				
	A	B	C	D
The successful innovation model of open source may be in danger. The fragile balance between intrinsically and extrinsically motivated developers may be disturbed by the entrance of extrinsically motivated commercial firms into the world of open source. [53]	0	0	-1	-1
The development of new payment policies will allow a far greater expansion of the willingness of engineers to participate in radical innovations. The incentive to solve remuneration issues should not be underestimated. [48]	0	-1	2	0
Expand and improve isn't exactly "innovation". [17]	0	1	-1	1
Open source is neither innovative nor new, at least insofar as its development model is concerned. [57]	-2	-1	2	0
The open source development model cannot be applied to the production of tangible-material products, and its applicability is thus mainly confined to the realm of software. [60]	0	0	-1	0

This is the discourse of the business world reaching out to the FOSS community: the entrance of commercial firms does not pose a threat to the community; rather, it contributes to the resolution of remuneration issues. And: the open source development model fulfils the requirements of the innovation process within a corporate context, where reducing costs and speeding up production is crucial. The open source model is neither innovative nor new, which is precisely why it can be applied to the production of material products: its effectiveness has been proven time and again.

The third discourse claims that innovation occurs at the interface between business firms and FOSS communities, and is characterised by imitation, geared to producing the same product in faster and cheaper ways. The FOSS development model enables such innovation by virtue of being a great imitator, rather than by bringing disruptive products to the market. Reflecting thus the orientation of the innovation process within a corporate context toward such functional requirements (as reducing costs and speeding up product development), FOSS presents a favourable opportunity for the business community to seize.

Discourse D: Cumulative cultural innovation

While the third discourse regards imitation as integral to the innovation process, the fourth discourse holds that innovation is essentially about a radical new idea.

Table 17				
	A	B	C	D
Expand and improve isn't exactly "innovation". [17]	0	1	-1	1
Code elegance is crucial to software innovation. [62]	1	3	0	-2

Innovation is about a new idea, hence 'you can hack a great innovative idea in a very poor way but the idea behind it is still innovative' (J. Burke, respondent #26, Roomware project developer), as 'software innovation is about an idea, and if users like it, they will use it regardless of how elegant the code is' (FOSS developer G-J. Braas, respondent #29). Besides, if the code is not elegant, one can still try to make it more elegant, but if the underlying idea is not innovative, that nobody can change.

Table 18				
----------	--	--	--	--

	A	B	C	D
Reverse engineering is mistakenly seen as important to innovation and competition in all industrial contexts. [12]	0	0	-2	-2
The GNU GPL stifles innovation and impedes reuse. Exactly the kinds of things you ostensibly wouldn't want an "open source" license to do. [5]	-2	-1	-3	-3
At best, software patents have thus far had a neutral effect on software innovation; at worst, they have had a significant negative impact on future innovation. And there is simply no evidence that patents are needed to promote innovation in software. [6]	2	3	3	3
Government should not develop innovation policies to fund open source communities as competitors to proprietary incumbents. [24]	-1	-1	-1	-3

Technological innovation, being inherently cumulative, thrives on practices of reverse engineering. Reverse engineering, the process of analysing a technology product to identify how it functions on the level of its components and their interrelationships, is what ensures that innovation is not held aback by technological black boxes. As Adriaan de Groot (respondent #12), contributor to the KDE project, puts it: 'Due to the volume of proprietary development, it is necessary to be able to reverse-engineer so that you understand'. This culture of curiosity and of tinkering with technology, is hampered by patents, and enabled by open standards and open software, which the government could fund in order to ensure their sustainability as public infrastructures.

Table 19				
	A	B	C	D
Standards inhibit innovation. That's not always a bad thing, because it allows software to work together, but at its heart, a standard's purpose is to stop you from creating you own way of doing something. [50]	-1	-2	-1	-3
Open standards and user freedom of choice allows innovation, it even promotes innovation. Users get to choose, which provides an environment of evolution and of survival of the fittest. [51]	3	1	1	3

In this discursive formation, open standards 'open up a market to innovation' (Adriaan de Groot, respondent #12), for they enable the creation of new products and innovations atop existing ones: 'for example, if the HTTP protocol was closed, nobody would be able to develop a Web browser' (Perl developer Timo Schreider, respondent #19). What proves the importance of open standards for innovation is the fact that interoperable applications are a source of innovation and interoperability is inconceivable without open standards. That is, as FOSS hacker Philip Poten (respondent #20) remarks, why 'everything which leads to a standard is an innovation, rather than inhibits innovation'.

Table 20				
	A	B	C	D
Although the open source method for producing software is an important organizational innovation, much of the software that is being produced under this method today is not innovative in the sense that VisiCalc (the first spreadsheet) or WordStar (one of the first word processors) was. Indeed, most open source projects under way are meant to imitate (or clone) existing for profit software. [21]	-1	1	1	-1
Open source principles could potentially be applied to almost any intellectual endeavor, and may be a very important innovation toward harnessing global collaboration toward solving complex public policy and management problems. [42]	2	-1	0	3

The fact that some FOSS projects re-implement functionality embodied in proprietary software means nothing: for the aim of most, if not all, software, including proprietary software, is to re-implement functions embodied in other products. The innovation of the open source development model consists precisely in creating new things out of existing ideas. As says Maarten Wijnen-Meljer (respondent #14), a consultant with the Programme for Open Standards and Open Source Software in (the Dutch) government (OSOSS), innovation 'is not about a new idea, but about implementing them in a way that can be used...Innovation and creativity are not so much about inventing ... but about having a purpose'.

The open source development model is not new, considering that practices of sharing have always been widespread among technologists and scientists, but it could provide the necessary 'tool-kit' for social innovation, for open source is 'an innovation in how people co-ordinate' ... 'a cultural innovation' (James Burke, respondent #26). The fourth discourse expands the field of the FOSS development model beyond the realm of software, as the work ethic of FOSS developers, which consists in working on what one wants in collaboration with others, is a powerful motive force to warrant application in contexts conducive to distributed development.

From this discourse springs forth the image that innovation thrives on open standards, for interoperability foment innovation, and open source is a cultural innovation, empowering a multitude of small groups to re-invent collaboration and co-ordination on a global scale.

■ Implications for innovation policy and management

By examining the dispersion of FOSS contributors' subjectivity across four discourses, we have arrived at four conceptions of innovation, namely (a) radical user innovation, (b) small-group innovation, (c) sustaining business innovation and (d) cumulative cultural innovation. This result firmly suggests that individuals with different perspectives on FOSS coalesce around FOSS projects, thus lending support to the findings of prior studies of FOSS developers' motivation, which have shown that FOSS projects are able to attract contributors with diverse motivations (Ghosh, 2005; Ghosh, *et al.*, 2002; Hertel, *et al.*, 2003; Lakhani and Wolf, 2005; Shah, 2006). In this respect, the dispersion of contributors' subjectivity across four discourses reflects the diversity of opinion characteristic of a community as heterogeneous as the FOSS community. The obvious implication for innovation policy and management is that the innovation process does not so much require a shared meaning among participants as a set of *enabling conditions* that encourage participation, collaboration and experimentation. Which are those enabling conditions? Our analysis shows that all discourses consider (a) generalised and unfettered participation by end users' and (b) the commons as key to unlocking innovation. But let us look more closely at the implications for policy and management that these commonalities highlight.

The contribution of users to the innovation process has been known for a long time, though its significance has often been overlooked. No historical example illustrates this better than the famous *Encyclopédie*, edited by D'Alembert and Diderot in the eighteenth century. Despite the tendency to associate the development of the *Encyclopédie* with such prominent historical figures as Diderot, D'Alembert, Rousseau and Voltaire, attributing thus its authorship to no more than a handful of gifted individuals, the writing of the *Encyclopédie* actually enlisted a multitude of artisans and craftsmen: 'in fact, anybody who could supply information of a technical kind, offered help in what became virtually a national effort' [17]. For a contemporary example, one need look no further than at how innovation in the capital goods sector — a major source of technical change over the past two centuries — is heavily dependent upon its users: for example, the participation of airlines in the innovation process stretches from the development of new product designs in collaboration with their suppliers to the post-production identification and elimination of bugs (Mowery and Rosenberg, 1988; Rosenberg, 1988a; see also Rosenberg, 1963a, 1963b). But user innovation is far from confined to producer goods. The case of radio amateurs illustrates succinctly users' inventiveness in dealing with practical technical problems:

In the early days of radio, amateurs were assigned the range of short-wave signals — less than 200 meters — precisely because the authorities thought that nothing much could be done ... As it turned out, clever amateurs, who did not know that nothing could be done, quickly demonstrated that effective transmission was possible ... Establishing precisely why this performance so vastly exceeded expectations led to major discoveries on the nature of the ionosphere. [18]

There is hardly a field of human activity untouched by the effect of user involvement. The artistic realm is no exception: by virtue of turning spectators into participants, Bertolt Brecht's 'epic theatre' is considered the most significant technical innovation in modern theatre [19]. It has been said that, in consequence of the increasing application of scientific knowledge to technical problems since the middle decades of the nineteenth century, scientific investigators and engineers were about to displace laborious amateurs in the innovation process [20]. But recent history refutes it: such innovations as computer games originate in the virtuosity of enthusiastic users rather than manufacturers [21].

Most important, users, by continuously reinventing the very games they are playing, constitute not only a long-lasting but the decisive source of innovation in the computer games industry [22]. The findings of our research point toward the same conclusion, for contributors to FOSS projects share the conviction that users are a more potent source of innovation than manufacturers, thus calling attention to the potentialities for radical innovation issuing forth from opening up the innovation process to the widest possible number of participants.

The commonalities among the discourses suggest that innovation is not the fruit of the profit motive. Software innovation in particular does not depend on the investment allowed for by commercial software but is independent of it. Recent research points to the same direction, as there is not any 'evidence that more market drives higher rates of innovation. Rather, the degree to which technological diffusion occurs via market exchange depends to a great extent on the nature of technological knowledge itself, *e.g.*, its degree of codifiability' [23]. Software can be easily digitised and distributed through the Internet. Consequently, it does not depend on the market for its diffusion across the social field [24].

Our findings also challenge the assumptions underlying the present framework of intellectual property rights. The argument that a broad and strengthened intellectual property regime has a positive effect on the incentives to innovate [25] is thrown into insignificance, as all discourses regard the GNU GPL, which establishes a *commons regime*, as an enabling incentive mechanism for innovation. This reinforces the conclusion of recent studies that intellectual property rights 'are not the most important device apt to profit from innovation; and second, they have at best no impact, or possibly even a negative impact on the underlying rates of innovation' [26].

Experimentation is the driving force of innovation. This has been long known. Whether it is the freedom to try out 'different experiments of living' (Mill, 2006) or economic experiments with respect both to technology and form and size of organisation (Rosenberg, 1992) — the freedom to experiment is fundamental to any society that wishes to make a serious commitment to innovation. But if harnessing experimentation is the crucial question, then the starting point of our reflections on the software commons should be the recognition of its role as a veritable laboratory of experimentation and innovation. For FOSS is indeed an experiment in social organisation based on a distinctive form of property: an example of 'how social organization can change the meaning of property, and conversely, how shifting notions of property can alter the possibilities of social organization' [27].

Long before the virtual commons and FOSS communities emerged, in the middle decades of the seventeenth century the Diggers proposed to 'dig' the common land of England — the original commons — with the intention of making it the basis of an alternative path of economic development (Hill, 1991). The commons, they argued, constitutes a laboratory of innovation, enabling new ways of looking at problems and solving them. However, as the Diggers were violently suppressed and reduced to a marginal position in the bosom of the Parliamentarians, while vast stretches of common land were privatised by acts of Parliament (the famous 'enclosures'), the idea of using the commons as a springboard of experimentation was doomed to atrophy for more than 300 years until it was rejuvenated in the late twentieth century by communities homesteading the electronic frontier. If indeed networks of distributed FOSS development hold lessons that extend beyond the software industry, as it has been argued [28], then what FOSS discourses have to say about the commons as the driving force of innovation and the participation of users in the innovation process might yield insights to a bewildering array of questions that are continually occurring in innovation-driven societies. From this vantage point, the reason for giving the freest scope possible to the development of commons-oriented projects — through inclusive property structures and infrastructures for generalised user innovation — becomes evident. **16**

About the authors

George Dafermos is Professor of Technology Policy and Management at the National Institute of Higher Studies (IAEN) in Ecuador and a member of the research team of the FLOK Society Project (<http://www.floksociety.org>). His research interests are centred on the governance of the Commons and on practices of peer production and open/user innovation. **Michel J.G. van Eeten** is Professor of Public Administration at the Faculty of Technology, Policy and Management, Delft University of Technology. He also teaches in several programs for executive education at the Netherlands School of Public Administration in the Hague. His recent research has been focused on the governance of infrastructures, most notably on the issue of Internet security. Corresponding author E-mail: georgedafermos [at] gmail [dot] com

Acknowledgments

An earlier version of this paper was presented at the 2008 EURAM Conference in Ljubljana, Slovenia. We are grateful for the helpful feedback received on that occasion. Also, we would like to thank Job van Exel and Gjalte de Graaf for their invaluable methodological assistance; Casper Harteveld, Emiel Kerpershoek, Harald Warmelink as well as the anonymous *First Monday* reviewers for their comments; and of course the Q-sorters whose perspectives supplied the material for this inquiry. Needless to say, all errors and omissions remain our own.

Notes

1. The simplest definition of a *hacker* is that of somebody who tinkers with computer software or hardware, modifying it in ways that are often not anticipated (or endorsed) by its original creators and manufacturers. A more elaborate definition springs from the *hacker ethic* (Himanen, 2001; Levy, 1984) — a set of principles operating as a normative framework to which the activities of persons identifying themselves as hackers must conform — which emphasises the political demand for freedom of information as well as hackers' opposition to authoritarian and bureaucratic structures. Hence, by contrast to the typical mass media portrayal of hackers as computer criminals, it follows that what defines a hacker has very little, if anything, to do with the legality of their actions.
2. Osterloh and Rota, 2007, p. 164.
3. GNU GPL is an abbreviation for GNU General Public License, which is the most widely used FOSS license.
4. Van Eeten, 2001, pp. 395-396; see also Dryzek and Berejikian, 1993.
5. 'The volume of statements on a topic was originally referred to as a population or trait universe [in Stephenson (1953)], but has been rechristened *concourse* [in Stephenson (1978)] to connote the running together of ideas in thought' (Brown, 2001).
6. The statements were collected from the authors' personal archive of research papers on FOSS and from World Wide Web search results returned by Google on 23 October 2006 for the words open, source, free, software, innovation (the first 260 results of about 4,040,000 results were consulted) and USENET/Google groups (the first 130 results of about 5,930 results were considered).
7. In the literature the Q-set is sometimes referred to as Q-sample. For the purpose of clarity, only the term Q-set is used throughout this paper.
8. Foucault, 2002, p. 55.
9. Van Exel and de Graaf, 2005, p. 6.
10. Brown quoted in de Graaf, 2003, p. 66.
11. The other six persons were interviewed in the periphery of the *23rd Chaos Communication Congress*, an international hacker conference which took place in Berlin, Germany in December 2006.
12. Although two researchers may derive two different Q-sets from the same *concourse*, that is not considered problematic. As Brown (1993) explains, 'first, the structure chosen is only a logical construct used by the investigator. Whatever the starting point, the aim is always to arrive at a Q set that is representative of the wide range of existing opinions about the topic. Second, irrespective of the structure and of what the researcher considers a balanced set of statements, eventually it is the subject that gives meaning to the statements by sorting them'.
13. Van Exel and de Graaf, 2005, p. 8.
14. Foucault, 2002, p. 54.
15. Hall, 2003, p. 44.
16. All respondents named in the study consented to have their names printed.
17. Tancock, 1976, p. 10.
18. Rosenberg, 1988b, p. 144.
19. Benjamin, 1986, p. 233.
20. Mumford, 1963, pp. 212-221.
21. The first computer game (*Spacewar*) was developed by Steve Russell at MIT in the early 1960s (Herman, *et al.*, 2002; Kuittinen, 1997).
22. Pearce, 2002; Prügl and Schreier, 2006, p. 239.
23. Dosi, *et al.*, 2006, p. 1,118.
24. Joseph Schumpeter, writing more than half a century ago, asserted that the entrepreneur is the essential carrier of innovation, which has been interpreted by later commentators as to imply that commercial success is the ultimate criterion of innovation. This view is still in vogue today. Characteristically, Dahlin and Behrens (2005) 'use the convention of labelling a new idea or product an *invention* if its market success is unknown; [and] use the label innovation when there is established commercialisation'. In our opinion, this definition of innovation is misleading for it confounds entrepreneurial savvy — which is nothing other than the ability to operationalise an idea — with market success. But while there is no doubt that entrepreneurial skills are of instrumental value to the innovation process, these skills are not monopolised by firms. As a study of four innovations in medical equipment technology observes, users 'play an entrepreneurial role as they establish and organize the required innovation networks' (Lettl, *et al.*, 2006).
25. *E.g.*, Arora and Gambardella, 1994; Arora, *et al.*, 2002, pp. 115-141.
26. Dosi, *et al.*, 2006, p. 1,110.
27. Weber, 2004, p. 1.
28. *E.g.*, Demil and Lecocq, 2006, p. 1,463; von Krogh and von Hippel, 2006, p. 982; de Laat, 2007, p. 174; Weber, 2004, p. 17>.

References

- A. Arora and A. Gambardella, 1994. "The changing technology of technological change: General and abstract knowledge and the division of innovative labour," *Research Policy*, volume 23, number 5, pp. 523-532.
doi: [http://dx.doi.org/10.1016/0048-7333\(94\)01003-X](http://dx.doi.org/10.1016/0048-7333(94)01003-X), accessed 20 November 2014.
- A. Arora, A. Fosfuri and A. Gambardella, 2002. *Markets for technology: The economics of innovation and corporate strategy*. Cambridge, Mass.: MIT Press.
- W. Benjamin, 1986. "The author as producer," In: W. Benjamin. *Reflections: Essays, aphorisms, autobiographical writing*. Translated by Edmund Jephcott; edited and with an introduction by Peter Demetz. New York: Schocken Books, pp. 220-238.
- Y. Benkler, 2006. *The wealth of networks: How social production transforms markets and freedom*. New Haven, Conn.: Yale University Press.
- P. Biernacki and D. Waldorf, 1981. "Snowball sampling: Problems and techniques of chain referral sampling," *Sociological Methods Research*, volume 10, number 2, pp. 141-163.
- S.R. Brown, 2001. "The history and principles of Q methodology in psychology and the social sciences," at <http://facstaff.uww.edu/cottlec/QArchive/Bps.htm>, accessed 20 November 2014.
- S.R. Brown, 1993. "A primer on Q methodology," *Operant Subjectivity*, volume 16, numbers 3/4, pp. 91-138, and at <http://www.operantsubjectivity.org/pub/244/>, accessed 20 November 2014.
- S.R. Brown, 1980. *Political subjectivity: Applications of Q methodology in political science*. New Haven, Conn.: Yale University Press.

- K.B. Dahlin and D.M. Behrens, 2005. "When is an invention really radical? Defining and measuring technological radicalness," *Research Policy*, volume 34, number 5, pp. 717-737. doi: <http://dx.doi.org/10.1016/j.respol.2005.03.009>, accessed 20 November 2014.
- B. Demil and X. Lecocq, 2006. "Neither market nor hierarchy nor network: The emergence of bazaar governance," *Organization Studies*, volume 27, number 10, pp. 1,447-1,466. doi: <http://dx.doi.org/10.1177/0170840606067250>, accessed 20 November 2014.
- G. Dosi, L. Marengo and C. Pasquali, 2006. "How much should society fuel the greed of innovators? On the relations between appropriability, opportunities and rates of innovation," *Research Policy*, volume 35, number 8, pp. 1,110-1,121. doi: <http://dx.doi.org/10.1016/j.respol.2006.09.003>, accessed 20 November 2014.
- J.S. Dryzek and J. Berejikian, 1993. "Reconstructive democratic theory," *American Political Science Review*, volume 87, number 1, pp. 48-60. doi: <http://dx.doi.org/10.2307/2938955>, accessed 20 November 2014.
- M. van Eeten, 2001. "Recasting intractable policy issues: The wider implications of the Netherlands civil aviation controversy," *Journal of Policy Analysis and Management*, volume 20, number 3, pp. 391-414. doi: <http://dx.doi.org/10.1002/pam.1000>, accessed 20 November 2014.
- B. Eich and D. Hyatt, 2003. "Mozilla development roadmap," at <http://www-archive.mozilla.org/roadmap/roadmap-02-Apr-2003.html>, accessed 20 November 2014.
- N.J. van Exel and G. de Graaf, 2005. "Q methodology: A sneak preview," at <http://qmethod.org/articles/vanExel.pdf>, accessed 20 November 2014.
- M. Foucault, 2002. *Archaeology of knowledge*. Translated by A.M. Sheridan Smith. London: Routledge.
- E. Franck and C. Jungwirth, 2003. "Reconciling rent-seekers and donators — The governance structure of open source," *Journal of Management and Governance*, volume 7, number 4, pp. 401-421. doi: <http://dx.doi.org/10.1023/A:1026261005092>, accessed 20 November 2014.
- R.A. Ghosh, 2005. "Understanding free software developers: Findings from the FLOSS Study," In: J. Feller, B. Fitzgerald, S. Hissam and K.R. Lakhani (editors). *Perspectives on free and open source software*. Cambridge, Mass.: MIT Press.
- R.A. Ghosh, R. Glott, B. Krieger, and G. Robles, 2002. *Free/libre and open source software: Survey and study*. Brussels: European Commission/Information Society Technologies.
- G. de Graaf, 2003. "Tractable morality: Customer discourses of bankers, veterinarians and charity workers," Ph.D. dissertation, *ERIM Ph. D. Series research in management*, number 31. Erasmus Universiteit Rotterdam.
- S. Hall, 2003. "The work of representation," In: S. Hall (editor). *Representation: Cultural representations and signifying practices*. London: Sage in association with the Open University, pp. 13-74.
- L. Herman, J. Horwitz, S. Kent and S. Miller, 2002. "The history of video games," at <http://www.gamespot.com>, accessed 20 November 2014.
- G. Hertel, S. Niedner and S. Herrmann, 2003. "Motivation of software developers in open source projects: An Internet-based survey of contributors to the Linux kernel," *Research Policy*, volume 32, number 7, pp. 1,159-1,177. doi: [http://dx.doi.org/10.1016/S0048-7333\(03\)00047-7](http://dx.doi.org/10.1016/S0048-7333(03)00047-7), accessed 20 November 2014.
- C. Hill, 1991. *The world turned upside down: Radical ideas during the English Revolution*. London: Penguin Books.
- P. Himanen, 2001. *The hacker ethic and the spirit of the information age*. London: Vintage.
- E. von Hippel and G. von Krogh, 2006. "Free revealing and the private-collective model for innovation incentives," *R&D Management*, volume 36, number 3, pp. 295-306.
- N. Iivari, 2010. "Discursive construction of 'user innovations' in the open source software development context," *Information and Organization*, volume 20, number 2, pp. 111-132. doi: <http://dx.doi.org/10.1016/j.infoandorg.2010.03.002>, accessed 20 November 2014.
- T. Jordan, 2009. "Hacking and power: Social and technological determinism in the digital age," *First Monday*, volume 14, number 7, at <http://firstmonday.org/article/viewArticle/2417/2240>, accessed 20 November 2014.
- G. von Krogh and E. von Hippel, 2006. "The promise of research on open source software," *Management Science*, volume 52, number 7, pp. 975-983. doi: <http://dx.doi.org/10.1287/mnsc.1060.0560>, accessed 20 November 2014.
- P. Kuitinen, 1997. "History of video games," on file with authors.
- P.B. de Laat, 2005. "Copyright or copyleft? An analysis of property regimes for software development," *Research Policy*, volume 34, number 10, pp. 1,511-1,532. doi: <http://dx.doi.org/10.1016/j.respol.2005.07.003>, accessed 20 November 2014.
- K.R. Lakhani and R.G. Wolf, 2005. "Why hackers do what they do: Understanding motivation and effort in free/open source software projects," In: J. Feller, B. Fitzgerald, S. Hissam and K.R. Lakhani (editors). *Perspectives on free and open source software*. Cambridge, Mass.: MIT Press.
- C. Lettl, C. Herstatt and H.G. Gemuenden, 2006. "Users' contributions to radical innovation: Evidence from four cases in the field of medical equipment technology," *R&D Management*, volume 36, number 3, pp. 251-272. doi: <http://dx.doi.org/10.1111/j.1467-9310.2006.00431.x>, accessed 20 November 2014.
- S. Levy, 1984. *Hackers: Heroes of the computer revolution*. Garden City, N.Y.: Anchor Press/Doubleday.
- P.J. McKenzie, J. Burkell, L. Wong, C. Whippley, S.E. Trosow and M. McNally, 2012. "User-generated online content 1: Overview, current state and context," *First Monday*, volume 17, number 6, at <http://firstmonday.org/article/viewArticle/3912/3266>, accessed 20 November 2014. doi: <http://dx.doi.org/10.5210/fm.v17i6.3912>, accessed 20 November 2014.
- M.B. McNally, S.E. Trosow, L. Wong, C. Whippley, J. Burkell and P.J. McKenzie, 2012. "User-generated online content 2: Policy implications," *First Monday*, volume 17, number 6, at <http://firstmonday.org/article/viewArticle/3913/3267>, accessed 20 November 2014. doi: <http://dx.doi.org/10.5210/fm.v17i6.3913>, accessed 20 November 2014.
- J.S. Mill, 2006. *On liberty; and, The subjection of women*. Edited by Alan Ryan. London: Penguin.
- D.C. Mowery and N. Rosenberg, 1988. "Technical change in the commercial aircraft industry, 1925-1975," In: N. Rosenberg (editor). *Inside the black box: Technology and economics*. Cambridge: Cambridge University Press, pp. 163-177.
- L. Mumford, 1963. *Technics and civilization*. San Diego: Harvest.
- M. Osterloh and S. Rota, 2007. "Open source software development — Just another case of collective invention?" *Research Policy*, volume 36, number 2, pp. 157-171. doi: <http://dx.doi.org/10.1016/j.respol.2006.10.004>, accessed 20 November 2014.
- P. Passell, 2003. "Why open source stifles innovation," *Strategy+Business*, at <http://www.strategy-business.com/article/8160?gko=cb492>, accessed 20 November 2014.
- C. Pearce, 2002. "Emergent authorship: The next interactive revolution," *Computers & Graphics*, volume 26, number 1, pp. 21-29.
- R. Prügl and M. Schreier, 2006. "Learning from leading-edge customers at *The Sims*: Opening up the innovation process using toolkits," *R&D Management*, volume 36, number 3, pp. 237-250. doi: <http://dx.doi.org/10.1111/j.1467-9310.2006.00433.x>, accessed 20 November 2014.
- E.S. Raymond, 1998. "The cathedral and the bazaar," *First Monday*, volume 3, number 2, at <http://firstmonday.org/article/view/578/499>, accessed 20 November 2014.
- N. Rosenberg, 1992. "Economic experiments," *Industrial and Corporate Change*, volume 1, number 1, pp. 181-203. doi: <http://dx.doi.org/10.1093/icc/1.1.181>, accessed 20 November 2014.
- N. Rosenberg, 1988a. "Learning by using," In: N. Rosenberg (editor). *Inside the black box: Technology and economics*. Cambridge: Cambridge University Press, pp. 120-140.
- N. Rosenberg, 1988b. "How exogenous is science?" In: N. Rosenberg (editor). *Inside the black box: Technology and economics*. Cambridge: Cambridge University Press, pp. 141-160.
- N. Rosenberg, 1963a. "Technological change in the machine tool industry, 1840-1910," *Journal of Economic History*, volume 23, number 4, pp. 414-443.
- N. Rosenberg, 1963b. "Capital goods, technology and economic growth," *Oxford Economic Papers*, New Series, volume 15, number 3, pp. 217-227.
- W. Scacchi, 2010. "Computer game mods, modders, modding, and the mod scene," *First Monday*, volume 15, number 5, at <http://firstmonday.org/article/viewArticle/2965/2526>, accessed 20 November 2014.
- C.M. Schweik and R. English, 2007. "Tragedy of the FOSS commons? Investigating the institutional designs of free/libre and open source software projects," *First Monday*, volume 12, number 2, at <http://firstmonday.org/article/view/1619/1534>, accessed 20 November 2014.
- S.K. Shah, 2006. "Motivation, governance, and the viability of hybrid forms in open source software development," *Management Science*, volume 52, number 7, pp. 1,000-1,014. doi: <http://dx.doi.org/10.1287/mnsc.1060.0553>, accessed 20 November 2014.
- W. Stephenson, 1978. "Concourse theory of communication," *Communication*, volume 3, pp. 21-40.
- W. Stephenson, 1953. *The study of behavior: Q-technique and its methodology*. Chicago: University of Chicago Press.
- S. Stevenson, 2010. "Unlimited potential, unlimited power? Microsoft's corporate citizenship in the battle over new social relations of production," *First Monday*, volume 15, number 10, at <http://firstmonday.org/article/viewArticle/3087/2626>, accessed 20 November 2014.
- S. Stevenson, 2007. "Public libraries, public access computing, FOSS and CI: There are alternatives to private philanthropy," *First Monday*, volume 12, number 5, at <http://firstmonday.org/article/view/1833/1717>, accessed 20 November 2014.

L. Tancock, 1976. "Introduction," In: D. Diderot. *Rameau's nephew; and, D'Alembert's dream*. Translated by L. Tancock. Harmondsworth: Penguin.

V. Valloppillil, 1998. "Open source software: A (new?) development methodology," Microsoft (unpublished memo), at <http://www.catb.org/~esr/halloween/halloween1.html>, accessed 20 November 2014.

S. Weber, 2004. *The success of open source*. Cambridge, Mass.: Harvard University Press.

G. Weston, 2004. "Message posted on comp.sys.mac.system Usenet newsgroup" (22 September).

J. Zawinski, 1999. "Resignation and postmortem," at <http://www.jwz.org/gruntle/nomo.html>, accessed 20 November 2014.

Editorial history

Received 21 November 2013; accepted 12 October 2014.

This paper is licensed under a [Creative Commons Public Domain License](#).

Images of innovation in discourses of free and open source software
by George Dafermos and Michel J.G. van Eeten.
First Monday, Volume 19, Number 12 - 1 December 2014
<http://journals.uic.edu/ojs/index.php/fm/rt/prINTERfriendly/4210/4176>
doi: <http://dx.doi.org/10.5210/fm.v19i12.4210>