The Basement Interviews
Open Source

Eric Raymond, President Emeritus and Co-Founder of the Open Source Initiative, speaks to Richard Poynder

Published 24th March 2006[1]

Eric Raymond was born in Boston, Massachusetts, in 1957, the oldest of five children. His father worked as a computer programmer for Sperry Univac, which meant that for the first thirteen years of his life Raymond lived in a succession of foreign countries, including Italy, Venezuela and England. On their return the family settled in Pennsylvania.

Raymond was an extremely bright child with a mild case of cerebral palsy, a combination that was to blight his childhood, and made him a victim of frequent bullying. This, coupled with the strict discipline he experienced during his Catholic education, was to make Raymond deeply suspicious of any kind of power, and turn him into an anti-authoritarian. As he puts it: "I had a lot of childhood experiences which demonstrated that the only times people used force was when they wanted to beat me up."

As a young man Raymond's interests included math, philosophy, and music. Having to constantly adjust to new schools, new languages, and new cultures, he had also developed what he calls his "alien anthropologist mode" — a frame of mind he was later to use to great effect as the self-elected "tribal historian and resident ethnographer" of the Open Source and Free Software movements.

As a freshman at the University of Pennsylvania Raymond was immediately marked out as a potential math prodigy. Having found school insufficiently stretching for his above average talents, however, he lacked the necessary discipline or emotional maturity to cope with the demands of an undergraduate course, and after suffering a "math burnout" left without a degree.

His time at Penn was not wasted however: while ducking classes Raymond taught himself programming, and for the next five years he worked as a programmer for a number of technology companies.

Eventually concluding that he "didn't fit into the conventional corporate framework", in 1985 Raymond changed direction to become an independent consultant, and freelance writer. His first book project, written while sitting out a year's retainer with his last employer, Rabbit Software, was *Portable C and*

[1] The interview took place in July 2005 (http://poynder.blogspot.com/2006/03/basement-interviews.html)

UNIX Systems Programming.[2] Released under the pen name J E Lapin[3] the book was published in 1987.

A close friend of Richard Stallman's in the 1970s, Raymond was an early convert to the Free Software Movement,[4] and contributed to Stallman's GNU Project,[5] although the extent and significance of that involvement is a source of disagreement between Raymond and Stallman today. Raymond also contributed to a number of other Free Software projects, including the Gosmacs editor.[6]

Becoming increasingly interested in hacker culture, in 1990 Raymond took over the Jargon File — a collection of hacker slang originally created at Stanford in 1975.

When the first copies of the GNU/Linux operating system began to circulate in the 1990s, therefore, Raymond was well placed to report on what he quickly saw to be the most significant achievement of the Free Software Movement to date. Indeed, the story he subsequently told about the development of the Linux kernel[7] has become the dominant narrative of the Open Source Movement.

Raymond concluded that what was most radical about GNU/Linux was not so much that it was the first free operating system, but that in developing the kernel Torvalds had invented a totally new method for creating software. Moreover, in doing so he had contravened the cardinal rule of software engineering.

Since the 1970s any programmer worth his salt had implicitly believed Brooks' Law[8]. This held that the only reliable way of developing software was to create small, centrally controlled, hierarchically structured teams of professional developers, and have them beaver away in monkish isolation —

[2] *Portable C and UNIX Systems Programming,* J E Lapin, Prentice Hall PTR, 1987

[3] aka rabbit

[4] The free software movement began in 1983 when Richard Stallman announced the GNU project. The goal of the movement is to give freedom to computer users by replacing software that has restrictive licensing terms with free software. http://en.wikipedia.org/wiki/Free_software_movement

[5] Richard Stallman started the GNU Project in 1983. It initiated the development of the GNU operating system, and began in January 1984. GNU is a recursive acronym for "GNU's Not UNIX", which was chosen because its design is UNIX-like, but it contains no actual UNIX code. The GNU system, combined with the third-party kernel Linux (to become GNU/Linux), is now one of the most widely used operating systems in the world. http://en.wikipedia.org/wiki/GNU

[6] The Gosling Emacs (Gosmacs) was an Emacs implementation written in 1981 by James Gosling, the inventor of the Java programming language. It was the first Emacs to run under UNIX, and some of the code was later used by Richard Stallman to write GNU Emacs.

[7] The Linux kernel was begun as a hobby in 1991by Finnish university student Linus Torvalds while attending the University of Helsinki. Linux was subsequently combined with various components developed by the GNU Project to create the GNU/Linux operating system.

[8] Brooks' law was stated by Frederick P Brooks as, "Adding manpower to a late software project makes it later." It appeared in his 1975 book *The Mythical Man-Month*, Addison Wesley, 1995 (Anniversary Edition). http://en.wikipedia.org/wiki/Brooks%27_law

a method that Brooks characterised as being akin to the way in which Reims Cathedral had been built.[9]

By contrast, Linux had been created by thousands of geographically distributed hackers collaborating over the Internet. With frequent releases, and constant user feedback, Linux had emerged, said Raymond, out of "a great babbling bazaar of differing agendas and approaches".

Brooks' Law implied that such an anarchic approach would have inevitably ended in chaotic disarray. Instead, contrary to all expectations, the developers of Linux had created a kernel that not only worked, but was technically superior. Raymond concluded that this success was based on the principle that "given enough eyeballs, all bugs are shallow".[10]

Keen to share his insights, Raymond penned *The Cathedral & the Bazaar* — an essay that not only offered a compelling explanation of how the Free Software Movement had enabled such a model to arise, but turned out to have articulated something that all hackers knew subliminally, but had never brought to consciousness.

By providing this "generative myth", says Raymond, he gave the community the necessary focus to capitalise on the model that Torvalds and the other Linux developers had accidentally created. The value in doing so, he explains, is that unless you "organise people's perceptions of isolated facts even the most innovative set of innovations may languish in the margins of the economy for a long time."[11]

It turned out that some "suits" also found Raymond's narrative compelling. In January 1998 Netscape released the source code for the Netscape browser, in the hope that a community effort could achieve what Netscape on its own could not: hold back Microsoft's approaching monopoly on the web browser market.[12] As the Netscape press release put it, the aim was "to harness the creative power of thousands of programmers on the Internet by incorporating their best enhancements into future versions of Netscape's software."

To Raymond's immense satisfaction, Netscape executives indicated that *The Cathedral & the Bazaar* had influenced their decision.

Dubbed by Raymond "a shot heard around the world", Netscape's move not only provided an unexpected PR fillip for Free Software, but raised the

[9] The Notre-Dame de Reims (Our Lady of Rheims) is the Cathedral of Reims, France, where the kings of France were once crowned.

[10] Raymond described this as Linus' Law, not least because it was based on something that Torvalds had said. More formally, Raymond expressed it thus: "Given a large enough beta-tester and co-developer base, almost every problem will be characterised quickly and the fix obvious to someone." http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s04.html

[11] *Analog,* June-July 2004, http://www.catb.org/~esr/writings/analog.html

[12] The original press release is here: http://wp.netscape.com/newsref/pr/newsrelease558.html

possibility that it could gain mindshare in corporate boardrooms as well as among hackers.[13]

Believing that a once-in-a-lifetime opportunity had arisen, Raymond concluded that it was essential to "re-brand" Free Software. His fear was that while Fortune 500 companies might be amenable to an ethically neutral argument about efficiency improvements, the Free Software Foundation's moralising about the ethical imperatives of making software free would only alienate them. In February 1998, therefore, Raymond flew to the West coast to meet with a group of like-minded people, and co-founded the Open Source Initiative.[14]

Raymond's message was simple: in a world in which software development is constantly growing in complexity small groups of isolated programmers working in secret can no longer deliver the goods. Today, therefore, success depends on making software code freely available, and allowing as many people as possible to "peer review" and test it. As he succinctly puts it "secrecy is the enemy of quality."

It was a powerful message, and throughout 1998 a growing roll call of influential software companies — including IBM, Sun Microsystems, Oracle, Informix, and Corel — announced initiatives to support Open Source. By the end of the year the Movement had acquired a sufficient head of steam that few could ignore it.

Success, however, has come at a price. Not only has Raymond sacrificed his friendship with FSF founder Richard Stallman, but the Movement has been split. While pragmatic Raymondism focuses primarily on marketing the concept of Open Source, idealistic Stallmanism insists that Free Software is an ethical issue; a matter of right versus wrong. By treating the issue as purely a question of efficiency, says Stallman, Raymondism "is not sufficient to give us freedom that is secure". In short, Stallman believes that since Raymondism lacks the conviction that Free Software is an end in itself, it threatens to subvert the aims of the Movement.

Like Stallman, therefore, Raymond has become a controversial figure. For Free Software groupies he is an egotistical pretender who has sold his community down the river. For the pragmatists he is a liberator who, in the words of Raymond's friend Jay Maynard,[15] has freed the Movement "from the straightjacket that Stallman tried to force on it."

[13] A hacker is a person who creates and modifies computer software and computer hardware, including computer programming, administration, and security. In computer programming, a hacker is a programmer who hacks or reaches a goal by employing a series of modifications to exploit or extend existing code or resources. In computer security, however, a hacker is a person able to exploit a system or gain unauthorised access through skill and tactics. Thus for many the word has taken on a negative connotation that earlier hackers like Raymond resent. http://en.wikipedia.org/wiki/Hacker

[14] http://en.wikipedia.org/wiki/Open_Source_Initiative

[15] http://www.tronguy.net/

What's the reality? We should not doubt that Stallmanist idealism could never have won over either the boardroom or Wall Street. Nor should we doubt, however, that Raymondism has introduced risks, not least the risk that the Movement could be co-opted by large corporations and exploited for their own ends — a danger all the greater in light of the Movement's internal split.

Perhaps the fundamental question the split raises is the extent to which the anarchic ethos of hackers can co-exist with hierarchical corporate culture. For Stallman the question is hardly relevant, since for him ethical issues are always prior. Raymond, however, believes that corporate culture can be gradually subverted. "In reality", he says, "it is we who are co-opting them. We are changing their culture, their products, and their way of doing things much more than they are changing us."

Herein, perhaps, lies a key issue for the wider free knowledge movement: should its aims be to replace traditional proprietary modes, or to establish peaceful co-existence with them? Can the world change dramatically, or will it simply become more variegated? More importantly, is there a risk that if they try too hard to co-exist with traditional proprietary models, the various free and open movements could be appropriated by them, and emasculated in the process?

##

Interviewing Stallman had felt like being mauled by a bad-tempered grizzly.[16] In preparing to interview Raymond, therefore, I was conscious that, in contrast to the hippy democratic principles espoused by Stallman, Raymond represents Second Amendment[17] libertarianism, and gun rights. Specifically, Raymond is a self-styled market anarchist who believes that citizens have the right to carry guns in order to protect themselves from the government. To a naïve European that seemed a little scary.

I also knew that in 1999 Raymond had famously sent a bruising e-mail threat to OSI co-founder Bruce Perens, who posted the message to a mailing list and indicated that — in the light of Raymond's gun habits — he had alerted the police.[18]

Nor, it seemed, could journalists assume that they would be spared Raymond's hard-man habits. ZDNet columnist John Carroll once characterised engaging with Raymond as akin to taking part in a café debate

[16] The Stallman interview can be read here: http://poynder.blogspot.com/2006/03/interview-with-richard-stallman.html

[17] Amendment II (the Second Amendment) of the United States Constitution, which is part of the Bill of Rights, declares the necessity for "a well regulated militia", and prohibits infringement of "the right of the people to keep and bear arms".
http://en.wikipedia.org/wiki/Second_Amendment_to_the_United_States_Constitution

[18] http://lists.debian.org/debian-user/1999/04/msg00623.html

where Raymond is "the guy at the table trying to take out his opponent's eye with a fork".[19]

Even Raymond's friends were warning me to tread carefully: "Eric cares about some things so deeply that it is hard to hold a rational conversation with him about them," Rob Landley told me.[20]

"If he thinks poorly of you, he'll tell you in great detail and at great length, sometimes whether you want to hear it or not," cautioned Maynard.

What kind of horrors awaited me in interviewing Raymond? I toyed with the idea of flying out to his lair in Pennsylvania, but wimped out, choosing the safer option of a telephone conversation.

In the event, Raymond was a pussycat. He was articulate, reasonable, and very friendly. He was also surprisingly frank, only evading one question. Even when I recklessly suggested that he had proved a Stalin to Stallman's Lenin he simply roared with laughter, replying sardonically: "Comparing me to Stalin! That is the nastiest thing that has happened to me in weeks."

Was it that the chemistry worked in my favour, or had Raymond decided on this occasion to play Mr Nice Guy? When I asked him if his reputation as an irascible interlocutor was justified he replied obliquely: "I invite you to judge by your interaction with me."

But how to judge? Is Eric Raymond the aggressive tub-thumper reaching for a fork to spear his opponent's eye, or the rational and eloquent essayist able to see and articulate the most pragmatic course of action?

Then there is Raymond the hacker; Raymond the gun-toting anarchist; Raymond the Sci-Fi fan; Raymond the "initiate Wiccan priest and coven leader of long standing"[21]; Raymond the teetotal polyamorist; Raymond the Tae Kwon Do Black Belt; not to mention Raymond the political cynic, for whom democracy "isn’t very interesting".

Once the interview started one thing became clear: it is difficult to have a conversation with Raymond without frequent references to RMS.[22] Indeed, it was hard not to conclude that Raymond has a little bit of a fixation on Stallman — a fixation clearly encouraged by the Open Source community which, says Landley, views Raymond as the "designated alternative to RMS".

Is this fixation mutual? Possibly. When I later e-mailed Raymond some comments Stallman had made about him he copied Stallman into his detailed and self-justificatory reply. This sparked a fascinating e-mail interchange

[19] *How the software economy is driven by proprietary work*, John Carroll, ZDNet , May 21st 2004 http://www.zdnet.co.uk/print/?TYPE=story&AT=39154927-39020682t-21000005c

[20] http://www.landley.net/

[21] NeoPaganism, Eric Raymond, http://wiccanhistorian.home.att.net/bos/neopaganism.html

[22] Stallman's entry in the first edition of The Hacker's Dictionary was: "'Richard Stallman' is just my mundane name; you can call me 'rms'". And to many he remains simply RMS.

between the two of them, into which I was copied. Judging by these e-mails both men remain very engaged with one another. Sadly, Stallman later decreed that the e-mail conversation had been "off the record".

Interview completed, however, the question remained: who is Eric Raymond? In *Revenge of the Hackers*[23] he talks of how, when talking to the press, he deliberately sets out to exude "attractive dissonance." To excite journalists, he explains, the trick is to "sound challengingly weird". To cheerfully discuss guns, anarchism, and witchcraft "while looking as well-groomed, boyishly charming, and all-American wholesome" as possible. He adds that it is important not to fake weirdness but to be genuinely weird.

Nevertheless, it's hard not to feel that much of Raymond's public persona is more spin than substance. At one point in the interview Raymond said that part of his job is to provoke thought, and get people to think about things. He added: "When that is your job description, if you are not pissing off a certain number of people a lot of the time then you are probably not pushing hard enough."

Undoubtedly Raymond has pissed off a lot of people in his time. On the other hand, his well-crafted and persuasive essays have also made a lot of people think.

In the end perhaps it doesn’t really matter whether or not Raymond is weird. In any case, I will leave readers to judge for themselves!

The interview begins…

RP: *Where and when were you born?*

ER: In Boston, Massachusetts, on the 4th of December 1957.

[23] *Revenge of the Hackers*, Eric Raymond, published in *The Cathedral & the Bazaar*, O'Reilly, 1999

RP: *You have siblings?*

ER: Yes. I am the oldest of five children.

RP: *Tell me about your father?*

ER: He was one of the earliest programmers, and worked on the electro-mechanical dinosaur computers of the early 1950s. He was employed by Sperry Univac,[24] which was the original computer company — a company that owed its origins to Eckert and Mauchly, the people who invented the modern digital computer. The company was later eclipsed by IBM, however, when its management went to sleep for a decade.

RP: *What sort of person was your father?*

ER: My father did high-altitude mathematical meteorology for the Air Force and was always very proud of the fact that when the Air Force sent its first draft of officers to MIT to study under Norbert Wiener[25] he was the only enlisted man in the group. Generally the only enlisted men to get that kind of opportunity — which was really reserved for officers — were those who were exceptionally bright and driven. That was my father. But he smoked and drank all his life and got throat cancer. We lost him in 2000.

RP: *How did you get along with him?*

ER: Boy that's complicated. He was very obsessive and a workaholic. I fought with him; I loved him a lot, and I learned a great deal from him. The rest of the family say I am a lot like him.

RP: *And your mother: she's a Catholic, which is why you went to a Catholic school?*

ER: Actually, my mother comes from a relaxed protestant background. It was my father who was Catholic. He wasn't really devout, but he thought that Catholic discipline was good for me and so sent me to a Catholic school.

RP: *Did your Catholic education influence your view of the world?*

ER: To some extent perhaps, but is hard for me to pin down, especially as I ended up being a neo-pagan[26] [laughs]. One thing my Catholic education did do, however, was to make me an anti-authoritarian.

[24] The American company UNIVAC began as the "business" computer division of Remington Rand formed by the purchase of the Eckert-Mauchly Computer Corporation in 1950. (EMCC was the company founded by, and named after, the two inventors/architects of the ENIAC.) UNIVAC is an acronym, standing for UNIVersal Automatic Computer. In 1955 Remington Rand merged with Sperry Corporation to become Sperry Rand. The UNIVAC division of Remington Rand was renamed the Univac division of Sperry Rand. http://en.wikipedia.org/wiki/Sperry_Univac

[25] Norbert Wiener was an American mathematician and founder of cybernetics. He created the term in his book *Cybernetics or Control and Communication in the Animal and the Machine* (MIT Press, 1948).

[26] Neopaganism or Neo-Paganism is any of a heterogeneous group of new religious movements, particularly those influenced by ancient, primarily pre-Christian and sometimes pre-Judaic religions.

***RP:** As a European, I'm curious to know how many generations your family has been in the US?*

ER: Both sides of my family have been in the country for a long time. My father's people were from central Pennsylvania — a little town called Crawford in the Appalachians. It was coal-mining country and they were pretty prosperous. In fact, if America had had the equivalent of the British landed gentry that would have been my father's family in the 19th Century. One was a pretty well known judge; another was a military office that fought and died at Gettysburg in the civil war. My grandfather was a railroad engineer back when that was a prestigious, hi-tech kind of job. But the family hit hard times during the Great Depression; he grew up poor, and it haunted him all his life.

***RP:** They were originally from Europe?*

ER: Right. My father's family were from Alsace-Lorraine, in France[27]; but they were probably ethnic Germans.

***RP:** And you mother's family?*

ER: We don't know where they came from originally, but we do know they were of Swiss German extraction. More recently they came from California, having gone through the classic "settlement of the West" story, arriving from Nebraska in a covered wagon.

***RP:** Your father's job meant that you travelled a lot during the first 13 years of your life?*

ER: Yes. At various times we lived in Venezuela, in Italy and in England. Basically, every couple of years we changed countries and I had to learn a new language. I really didn't like losing all my friends every time we moved, but it was very broadening, and gave me a perspective on the world that I think a lot of Americans don't have.

Never got along with the school system

***RP:** Your family later returned and settled in Pennsylvania. What are your memories of school?*

Often these are Indo-European in origin, but with a growing component inspired by other religions indigenous to Europe, such as Finno-Ugric, as well as other parts of the world. As the name implies, these religions are Pagan in nature, though their exact relationship to older forms of paganism is the source of contention. http://en.wikipedia.org/wiki/Neopaganism

[27] Alsace-Lorraine was historically a source of dispute between the nation states of France and Germany. The territory — which is composed of Alsace and parts of Lorraine — originally belonged to the Holy Roman Empire, but was gradually ceded to France. In 1871, after the Franco-Prussian War, it became part of the newly-founded German Empire, but was restored to France after World War I in the 1919 Treaty of Versailles. The area was then annexed by Nazi Germany in 1940, and again reverted to French control in 1945. http://en.wikipedia.org/wiki/Alsace-Lorraine

ER: I never really got along with the school system and I never went to a school that I liked.

RP: *This was because of your antipathy to authority?*

ER: Partly that; partly the fact that school systems are not designed for people as bright as I am. I'm sorry if that sounds immodest, but there is a certain range of IQs that schools are optimised for, and if you are either less bright than the lower end of their range, or brighter than the upper range, you have a hard time.

RP: *In one of your essays you say, "I was a skinny, runty kid with cerebral palsy and few social skills, fanatically devoted to improving my mind because it was the only part of me that seemed to work right."* [28] *While you have a slight limp, cerebral palsy has never been a serious handicap,*[29] *has it?*

ER: You can judge how serious it is by the fact that I have a black belt in Tae Kwon Do and do martial arts training. No, it is more of a cosmetic impairment than anything. It was much harder to cope with when I was a child. Anything that makes you visibly different can make life among children difficult.

RP: *Indeed, in his book on the Open Source Movement*[30] *Glyn Moody says that you had a hard time at school. Were you bullied?*

ER: Yes, that happened.

RP: *Physically bullied?*

ER: The whole range — social, verbal, and physical abuse.

RP: *Was this because of your cerebral palsy, or because you were brighter than them?*

ER: Both.

RP: *What were your interests at school?*

ER: The same things I am interested in today: theoretical mathematics, music, science, and linguistics, and philosophy — especially analytical philosophy.

RP: *You later went to the University of Pennsylvania to study math and philosophy?*

ER: Right. And then I hit an interesting wall: as the school system had never challenged me I had never learned how to study. This meant that when I got to college the results weren't pleasant, as I actually had to exert myself for the first time.

[28] *Dancing with the Gods*, Eric Raymond http://www.catb.org/~esr/writings/dancing.html

[29] Cerebral palsy or CP is a group of permanent disorders associated with developmental brain injuries that occur during foetal development, birth, or shortly after birth. It is characterised by a disruption of motor skills, with symptoms such as spasticity, paralysis, or seizures. http://en.wikipedia.org/wiki/Cerebral_palsy

[30] *Rebel Code*, Glyn Moody, Penguin Press, 2001

RP: *You left university without a degree, after what you have described as a math burnout?*

ER: Yes. There was a group of well-meaning professors at the University of Penn who thought I was a mathematical prodigy. They may have been correct — indeed there is some evidence that they were correct — but I didn't have the discipline or emotional maturity to match my talents, or pursue all the courses, so I flunked out.

RP: *So how did you spend your time at Penn?*

ER: I learned a lot of things, and picked up computer programming.

RP: *Given your father's profession, it's interesting that you didn't start programming until your undergraduate days.*

ER: True. I was exposed to computers all my life. I learned to play with a computer and think about computers at a time when it was really very, very rare — long before the personal computer revolution in the 1980s. One of my early memories — this must have been around 1967 — is of my father taking me in to Univac's benchmark computer centre in Rome and letting me play with an $8 million mainframe.

Computers always seemed like something I could do: it was just that mathematics captured my attention for a long time. So in the end programming was something I found for myself.

RP: *And computers became your chosen career path. It turned out, however, that working as an employed programmer was not for Eric Raymond, and in 1985 you went freelance. Your friend Rob Landley — who was technical editor for your book on UNIX Programming*[31] *— commented to me: "The last time Eric had a show-up-in-person 9-5 day job (rather than writing, telecommuting contracting work, and so on) was some place called Rabbit something or other, and he basically didn't like it." What didn't you like?*

ER: Actually, my experience at Rabbit Software was mixed. Some parts of the job I really liked, but I didn't fit into the conventional corporate framework much better than I had fitted into the school system.

RP: *What was your role there?*

ER: It was a start-up company doing communications software. I took the job because ever since I had read the original UNIX paper in *C.ACM*[32] I had been very interested in UNIX. Rabbit said they were planning to move to UNIX as their primary development system and I wanted to be a UNIX guru. So that's what I was for two and a half years.

[31] *The Art of UNIX Programming*, Eric Raymond, Addison-Wesley, 2003

[32] *Communications of the ACM,* D. M. Ritchie and K. Thompson, *17, No. 7 (July 1974), pp. 365-375* http://cm.bell-labs.com/cm/cs/who/dmr/cacm.html

Parse the jargon

***RP:** I guess an important moment for you occurred in 1990, when you became maintainer of the Jargon File[33] — now also published as The New Hacker's Dictionary.[34] The Jargon File had been dormant for seven years. Why did you take it on?*

ER: I first encountered the Jargon File in the later 1970s, and I thought it was really interesting and cool. It was also the first hint I had that Internet hackers — people like me — actually had a culture: that there was a community of minds out there. I also found it interesting because I have always had a tendency to flip into alien anthropologist mode, and the Jargon File offered a good way of practising that with hacker culture.

***RP:** What is "alien anthropologist mode"?*

ER: It's a characteristic I have that probably came from my early experience of living in different countries. Since I had had to live in so many cultures and speak so many languages when I was younger, I developed a set of reflexes for adapting to social situations that I don't understand at all well. It was a mindset where I thought: "Here I am, the anthropologist from Mars trying to be completely analytical and objective about what I am seeing." In doing so I tend to notice what the behavioural patterns around me are, and what the social patterns are. And I tend to apply that perspective everywhere I go. So it was natural that I should study the behaviour of the hackers around me in the same way.

***RP:** When did you first come across Free Software?*

ER: The answer to that depends on your perspective: I first came into contact with Internet hackers in the 1970s, and at that time there was a prevailing ethic of software sharing that was largely unconscious; it was just the way you behaved. Nobody really meditated about it, reflected on it, or issued manifestos about it. By 1982 I was participating in that ethic too, and developing what was called Free Software. I did some code for the Gosling Emacs editor,[35] for instance.

***RP:** This was already an online community, was it?*

ER: Right. The medium of the very early online community was the early ARPANET[36] and Usenet.[37]

[33] The Jargon File is an online collection of hacker slang from technical cultures including the MIT AI Lab, the Stanford AI Lab (SAIL), and others of the old ARPANET AI/LISP/PDP-10 communities http://www.catb.org/jargon/html/

[34] *The New Hacker's Dictionary*, Eric Raymond, MIT Press, 1996

[35] The Gosling Emacs (Gosmacs) was an Emacs implementation written in 1981 by James Gosling, the inventor of the Java programming language. It was the first Emacs to run under UNIX, and some of the code was later used by Richard Stallman to write GNU Emacs.

[36] ARPAnet had been created by the US Advanced Research Projects Agency (ARPA) in 1969. One of ARPA's goals was to connect mainframe computers at different universities around the country so that they would be able to communicate using a common language and a common protocol. In the 1980s

RP: *Were you writing Free Software as part of your day job, or was it a hobby activity?*

ER: It was intimately connected with my day job, and with the tools that I and the people at Rabbit Software used to do our jobs. I brought Gosmacs into the organisation and became the Gosmacs reference person for example. There were also things that would come up internally where Gosmacs needed to be customised, fixed, or extended, so I would make those changes and automatically send them back to the Gosmacs maintainers.

RP: *You say Free Software was part of hacker culture. Where did it originate?*

ER: That is a complicated question. Partly it stemmed from the few exceptional microcultures of collaborative development, including the MIT AI Lab,[38] and the Linpac Scientific Library, where some kind of prehistoric pre-open-source culture existed. Apart from those isolated microcultures, software was all proprietary. This is not surprising as initially programming had to be organised by wealthy organisations that could afford to buy the capital equipment and computers that were necessary.

As computing power became more available, and as communications costs dropped, however, more programmers were able to do programming for fun, and they picked up on the experience of the microcultures, and generalised from that.

RP: *So Free Software developed as a result of the culture of these microcultures being propagated more widely?*

ER: That is one way to look at it. But it would be wrong to imply that there was this Free Software Eden in which most software was free and then suddenly it wasn't. The myth of an early struggle between free and proprietary software was propagated by Richard Stallman, who over-generalised from his own exceptional experience at MIT.

The fact is that in the early days we all worked on proprietary software and proprietary platforms; and we accepted that: indeed there was no real concept that you could have an entire software stack that was free of trade secrets and other intellectual property restrictions. One of the things that Stallman deserves credit for is being brave enough to imagine such a possibility.

the ARPAnet gradually merged into the Internet, and was itself closed in 1994. http://en.wikipedia.org/wiki/History_of_the_Internet

[37] Usenet is a distributed Internet discussion system that was conceived by Duke University graduate students Tom Truscott and Jim Ellis in 1979. Users read and post email-like messages to a number of distributed categorised newsgroups. The medium is distributed among a large number of servers, which store and forward messages to one another. Individual users download and post messages to a single server, usually operated by their ISP or university, and the servers exchange the messages between each other. http://en.wikipedia.org/wiki/Usenet

[38] The MIT Artificial Intelligence Laboratory was an interdisciplinary research entity at MIT which became one of the most influential and accomplished in the fields of artificial intelligence and robotics. Research at MIT in the field of Artificial Intelligence began in 1959. In 1963, the (then) "AI Group" was incorporated into the newly formed Project MAC, only to split off again in 1970, as the MIT Artificial Intelligence Laboratory. Richard Stallman worked there in the 1970s. http://en.wikipedia.org/wiki/MIT_AI_Lab

***RP:** Stallman also took the practical step of starting the GNU Project in order to realise his dream?*

ER: He did. I have known Richard since the 1970s, and at one time he and I were close friends. So I was the first person he told about his plans for the GNU Project, and I was one of the earliest contributors to it.

***RP:** You have previously said that free and open-source software owes a lot to academic culture. Since a lot of hacking took place in universities and academic institutions, that makes sense.*

ER: Oh, absolutely. And the culture of peer review that eventually developed in the Open Source world is continuous with the culture of transparency historically prevalent in academia.

***RP:** What other factors helped to foster the Free Software Movement?*

ER: A really important input was UNIX. One of the things that hacker culture inherited from the UNIX guys was an obsession with software modularity and clean and properly layered designs. That aspect of modern hacker culture did not come from MIT, where by modern standards their architectures were ugly monoliths, with a lot of them written in assembly language.[39] Nor did it really come from computer science and academia, but from the UNIX guys, who had an engineering tradition of working on a fairly large scale in high-level languages with clean architectures.

***RP:** Why would that encourage a culture of Free Software?*

ER: Because without knowing how to do software architecture and modularity,[40] individuals, and even groups of people collaborating over the Internet, simply can't write software effectively at the scale required. Essentially, collaboration can't work without a really strong culture of software modularity, well-defined APIs,[41] well-defined layers, and well-defined functional separations.

***RP:** And modulation allows separate groups to work on individual software components, and then pool their efforts?*

ER: Correct.

***RP:** And this in turn encourages a culture in which software is swapped and shared in a non-proprietary manner?*

[39] A human-readable notation for the machine language that a specific computer architecture uses. http://en.wikipedia.org/wiki/Assembly_language

[40] Modularity is the property of computer programs that measures the extent to which they have been composed out of separate parts called modules. Programs that have many direct interrelationships between any two random parts of the program code are less modular than programs where those relationships occur mainly at well-defined interfaces between modules. http://en.wikipedia.org/wiki/Modularity_(programming)

[41] An API (application program interface) is a set of routines, protocols, and tools for building software applications. A good API makes it easier to develop a program by providing all the building blocks. A programmer puts the blocks together.

ER: Right.

Linux

RP: *The aim of the GNU Project was to create an entirely free operating system. However, the central component — the Linux kernel — was eventually developed outside the GNU Project, by an independent hacker in Finland named Linus Torvalds.*[42] *Linux was then combined with the various GNU components to create the first free operating system. When did you personally discover GNU/Linux?*

ER: In 1993. However, as I said, I was aware of the GNU Project, and an early contributor to it. I was also aware of Linux because a lot of people were talking about it on Usenet. But there is a difference between being aware of a bunch of isolated Free Software projects that are generally dependent on a proprietary platform to function, and suddenly being presented with an entire software stack free from the bottom up. So when I was sent the first distribution[43] of the new operating system, there was for me immediately a message about autonomy, and there was a message about technical quality — because I instantly noticed that this was really good stuff: it was a high-quality operating system.

RP: *You have said that the real surprise for you was the way that Linux had been developed. As an advocate of Brooks' Law*[44] *you believed that good-quality software could only be written by a small group of centrally controlled professional developers; Linux by contrast had been created by thousands of hackers collaborating over the web.*

ER: I was not necessarily an advocate of Brooks' Law, but like most software engineers I tended to believe it. Suddenly I realised that all that we knew about software engineering had to be wrong, or at least re-evaluated under new conditions that I didn't immediately understand.

[42] Linus Torvalds is the Finnish software engineer best known for initiating the development of Linux. http://en.wikipedia.org/wiki/Linus_Torvalds

[43] Before the first Linux distributions (or "distros"), a would-be Linux user was required to be something of a UNIX expert, not only knowing what libraries and executables were needed to successfully get the system to boot and run, but also important details concerning configuration and placement of files in the system. A Linux distribution consists of a UNIX-like operating system comprising the Linux kernel and other assorted Free Software/open-source software, and possibly proprietary software. Commercially backed distributions such as Red Hat, Ubuntu (backed by Canonical Ltd), SUSE (backed by Novell) and Mandriva, and community projects such as Debian and Gentoo assemble and test the software before releasing their distribution.

[44] From Fred Brooks 1975 book *The Mythical Man-Month*, Brooks' Law states that programming work performed increases with direct proportion to the number of programmers (N), but the complexity of a project increases by the square of the number of programmers (N^2). Therefore, it should follow that thousands of programmers working on a single project would become mired in a nightmare of human communication and version control. http://en.wikipedia.org/wiki/Brooks'_law

RP: *So you decided to conduct an experiment with Fetchmail*[45] *— exploiting the same development techniques as Torvalds had used to create Linux?*

ER: Yes. I ran Fetchmail as a conscious experiment, and for the next three years I set about trying to understand exactly how the Linux people were getting away with violating Brooks' Law. In thinking about it I was led to much larger and more interesting conclusions.

RP: *And you outlined these in your 1997 essay The Cathedral & the Bazaar.*[46] *Your main point was that Torvalds had challenged the traditional assumption that software has to be built like a cathedral "carefully crafted by individual wizards or small bands of mages working in splendid isolation" because the Linux community had demonstrated that a coherent and stable system could be produced by thousands of geographically distributed programmers working in a non-hierarchical fashion — a model you described as resembling "a great babbling bazaar of differing agendas and approaches". You concluded that quality was possible in this model because, as you put it, "given enough eyeballs, all bugs are shallow".*

ER: That's right.

RP: *Torvalds was not conscious that he was creating a new development model. Do you think that this new way of creating software was inevitable in a world where the network has become ubiquitous?*

ER: Absolutely. I have believed that from the moment I first became conscious of what was happening.

RP: *In his book The Success of Open Source,*[47] *Steven Weber discusses in some detail how the BSD version of UNIX*[48] *was developed. In fact, he concludes that it was actually Bill Jolitz*[49] *(not Linus Torvalds) who launched the first UNIX-based operating system for personal computers to be distributed over the web.*

ER: That is correct.

RP: *He also says that it was the BSD community that first developed the distributed development model that Linus is credited with, although he clearly recognises that Linux has proved more important in terms of its scale and success.*

ER: But there was a key difference. The BSD model insists on distribution-level control — the BSD people view it as an important feature that the entire system can

[45] Fetchmail is a utility found on some UNIX-like systems used to retrieve e-mail from a remote POP3, IMAP, ETRN or ODMR mail server to the user's local system. It was developed from the popclient program. http://en.wikipedia.org/wiki/Fetchmail

[46] *The Cathedral & the Bazaar*, Eric Raymond, O'Reilly, 1999

[47] *The Success of Open Source*, Steven Weber, Harvard University Press, 2005

[48] 386BSD, also known as JOLIX, is a free operating system produced from the BSD derived UNIX operating systems for the Intel 80386.http://en.wikipedia.org/wiki/386BSD

[49] William Frederick (Bill) Jolitz, commonly known as Bill Jolitz, co-wrote 386BSD in 1989 along with Lynne Jolitz. http://en.wikipedia.org/wiki/386BSD

be built from one big make. Linus' approach was different; he controlled only the kernel and left distribution building to others.

The political result of the BSD choice is that if you disagree with any aspect of a BSD-distribution's policy, you have to start an entire new distro and maintain it. In consequence, the BSD world is very fissiparous.

***RP:** How does that compare with Torvalds' model?*

ER: The Linux way means (a) every distro can use the same kernel, and (b) if you disagree with somebody's distro-level policy, you can install a replacement package. The Linux way is both technically and politically more loosely coupled; thus, it's more robust and scales better.

Another important difference is the timescale of releases: the BSD people never embraced "release early, release often" with the fervour the Linux culture did. This turns out to matter.[50]

***RP:** Weber also argues that it is not entirely accurate to describe the open-source development method as being like a bazaar, since most open-source projects (not least Linux) are actually quite hierarchical, and indeed Linux has become increasingly hierarchical, as its scale has grown.*

ER: He has mistaken the trees for the forest.

***RP:** How do you mean?*

ER: There are at least two confusions here. One: the presence of a well-defined leadership hierarchy on a project by no means implies that the many-eyeballs effect stops working. Only if the leadership stopped paying attention to corrections generated outside itself would that happen.

Two: the fact that individual projects develop a leadership hierarchy does nothing to change the way projects compete for attention from developers, and from distribution packagers.

At both levels, if you make bad choices (including over-structured or under-structured leadership) that would impede the bazaar effect from working, you get fairly rapid feedback about it — both intended feedback from developers and observable consequences in code quality. So these problems tend to be self-correcting.

***RP:** What the new model inevitably assumes, of course, is a world in which software code is free for anyone to view. You tend to emphasise the way in which software can be developed once it is free, but I'm wondering how integral to your thinking are the principles of making it free in the first place — the principles outlined by Stallman and the Free Software Foundation (FSF)?*

[50] Another central tenet of *The Cathedral and the Bazaar* is that when developing Open Source programs developers should release new versions more quickly than closed source development. As Raymond put it, "Early and frequent releases are a critical part of the Linux development model." http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s04.html

ER: That also is complicated. While I loved working in the Free Software community, and I liked the idea of Free Software, Stallman's views on how intellectual property was evil didn't interest me. What I knew I did want was the results of the project, and it was for that reason that I contributed to the GNU Project.

RP: *So it was the way in which software could be developed once the source code was freely available that interested you, not the ethical issues associated with making code available?*

ER: What was important to me were the high-quality results, the process transparency, and what an economist would call the low transaction costs for innovation that become possible when source code is freely available.

Like Richard Stallman, I'm interested in the ethical issues around software. Unlike him, I tend to answer the question "What is ethical?" in terms of economics and game theory. That is, by asking "What is economically sustainable? What kinds of behaviour lead to productive long-term cooperation among selfish agents?"

Until I understood how the economics of Open Source worked, I could only consider talk of the "ethics" around it to be wishful thinking or ungrounded idealism. Once I did understand the economics, I could tie various ethical claims to the self-interest of software developers, consumers, and third parties, and make sense of them.

Shot heard around the world

RP: *Let's come back to that in a moment. Certainly The Cathedral & the Bazaar immediately attracted a lot of attention within the hacker community. More importantly, perhaps, it gained mindshare outside the community. Specifically, in January 1998, Netscape announced that it was releasing the source code for its web browser. In doing so, Netscape executives indicated that your essay had influenced them. In what I take to be a reference to the American Revolution, you described the event as a "shot heard around the world". What was the significance of Netscape's decision, both to the Free Software Movement and to you personally?*

ER: It was important to the Movement because Netscape throwing open the source code was something that Wall Street would have to notice. As such it was a chance for us to break out of the ghetto that we had been in for years, and years: it raised the possibility that mainstream culture was finally going to take Free Software seriously (the term Open Source hadn't been invented at that point).

For me personally it was very clear that if I did not get out of the way and hide, I was going to become famous [laugh]. So I had to make a very personal decision about whether to run away, or to become a public advocate for the new way of doing things.

RP: *As we now know, you chose not to hide, but in February 1998 flew to the West Coast to meet with some like-minded people and founded the Open Source Initiative*

(OSI).[51] *The aim was to "re-brand" the Free Software Movement as Open Source. Why?*

ER: We needed to make the narrative around Free Software — and the propaganda about it — less off-putting to the mainstream, and to sell it to Wall Street.

RP: *Why did you want to sell the idea to Wall Street?*

ER: Because I am an idealist in my goals, but ruthless and pragmatic in my methods [laughs]; and I had figured out early on that if you want to change the world, one of the things you need to be able to do is persuade the people who can write large cheques: you need to go where the money and the power is.

I also realised that if you want to change people's behaviour — and business practices — the most effective way to do it is to present an ethically neutral efficiency argument. The question, therefore, was how to come up with a narrative — a brand if you like — that would be persuasive to people who have money and power.

RP: *So how would you explain to "people who have money and power" why Open Source is a good thing? Why, for instance, might they support it, or indeed invest in it?*

ER: Open Source is a development method that you get when you realise that secrecy is the enemy of quality, and decentralised peer review is its best friend. What we have learned in the most painful possible way is that if you don't institute de-centralised peer review in order for your code to be checked for bugs, most of the time what you end up with is crap. So you have to have transparency of process, and you have to have third parties checking your work. Without that you don't have any self-correction in the system.

RP: *In other words, Open Source developers are able to produce better software than those working in a closed source environment?*

ER: Absolutely. One of the most consistent results from studies of large-scale software engineering is that over 70% of projects are failures. That is a truly lousy success rate, and it says that there is something deeply wrong with the way we have traditionally been doing things.

RP: *And to outline the business case for Open Source, you wrote your essay, The Magic Cauldron?*[52]

ER: Right. That was the economics and game theory paper.

RP: *One point you make in The Magic Cauldron is that everyone has been treating the software industry as a manufacturing industry, whereas it is more logical to view*

[51] The "open source" label came out of a strategy session held in Palo Alto in reaction to Netscape's January 1998 announcement of a source code release for Navigator (as Mozilla). Amongst those attending were Todd Anderson, Larry Augustin, John Hall, Sam Ockman, Christine Peterson and Raymond. http://www.opensource.org/docs/history.php

[52] *The Magic Cauldron*, Eric Raymond, published in *The Cathedral & the Bazaar*, O'Reilly, 1999

it as a service industry. You suggest, therefore, that instead of relying on repeat sales, software companies should consider making their software freely available, and then sell services around it. Significantly, you do not claim that software should never be proprietary, but merely map out the benefits of making it open.

ER: Yes. I tried to outline the incentives that make it economically a better deal to cooperate with Open Source projects rather than to seek to resist them. Actually, in that paper I also talk about the circumstances under which closed source makes economic sense. They're unusual, but they do exist.

Free or open?

RP: *Rebranding Free Software as Open Source was a controversial move, and has divided the Movement. It has also created some confusion as to how Open Source software differs from Free Software. It might be helpful here if you explained the difference. There are, for instance, ten Open Source definitions.*[53] *How do these differ from the FSF's four freedoms?*[54]

ER: [long pause, a whistle]. Good question. The four freedoms are a statement of intention, and have the character of an ethical program. The Open Source definition is an attempt to implement that ethical program, and defines in a legalistic way the constraints that Open Source software licences must satisfy in order for those four freedoms to be fulfilled. So you could say that the four freedoms are strategy and the Open Source definition is tactics.

I'll add, though, that I believe in a Freedom Zero that Richard doesn't. Freedom Zero is this: programmers (and creative people in general) have a right to issue their creative work under any licence they choose — closed, open, or purple with pink polka-dots — and have that choice respected. If you don't like the licence a programmer is offering, you are free not to use his code, but not to attack his right to choose his own licence.

RP: *Would it be fair to say, then, that Open Source software is Free Software by any other name?*[55]

ER: I think that is correct. The difference between Open Source and Free Software is not a difference in what kind of software gets produced, or the working methods used, or even the people who do it: the difference lies in what propaganda surrounds it. I should add that the FSF did make a minor technical change to the Free Software

[53] http://www.opensource.org/docs/definition_plain.php

[54] In his interview with me Stallman articulated the Four Freedoms of Free Software as: the freedom to run a program as you wish; the freedom to study the source code and change it to do what you wish; the freedom to make copies and to distribute them to others; and the freedom to publish or, more generally, distribute modified versions.

[55] I.e. a thing is what it is, not what it is called. As in Shakespeare's *Romeo and Juliet*: "A rose by any other name would smell as sweet".

definition in 1999, but so far as I am aware there are currently no licences that are considered Open Source that are not also considered Free Software licences.

***RP:** So essentially when you talk about free and Open Source software you are talking about the same product but a different marketing message?*

ER: In effect. However, if you told the Free Software people that they were propagating a marketing message, they would hurt you a lot!

***RP:** I'm curious because, when I spoke to Stallman, he seemed to imply that Free Software is not necessarily the same thing as Open Source software.*

ER: Yes. Stallman and a few people around him are very unyielding in their insistence that they are not Open Source developers, and that Free Software is something fundamentally different. But the community clearly doesn't agree with them on this. This is expressed in people's behaviour: developers freely cross over between working on projects that are designated Open Source, and projects that are designated Free Software. People don't refuse to do the work because of political labels.

***RP:** Why does this ideological gap exist?*

ER: Essentially, the FSF and the pragmatist wing — with which I am associated — reflect two different assumptions about how best to change the world. Richard is an idealist. He believes that the necessary way to change the world is to change people's thoughts, and trust that their actions will follow. So he is focused on moral argument and persuasion aimed at changing people's thinking. His belief is that once they have assimilated his view of the world they will change their behaviour to match.

***RP:** What is the pragmatist's approach?*

ER: The line of propaganda characteristic of the pragmatist — which is the one I more or less pioneered — works in the other way. It says, "Give people a good reason to change their behaviour and their thoughts will follow." So OSI deliberately focuses on changing people's behaviour, giving them incentives to change their behaviour, and then being relaxed as to what kind of thoughts they have when their behaviour is changed.

***RP:** So for you the ethical issues are not relevant?*

ER: No, they are relevant. They are just not something you want to put in your public propaganda. Sometimes the best thing you can do for freedom is to shut the hell up about it.

***RP:** But if it means prioritising commerce over ethics, does not the pragmatist's approach risk a gradual erosion of the primary objective, which I take to be maximising the amount of Free Software available? Stallman appears to believe this has already happened. He commented to me: "Somebody had the idea of adding non-free programs to the GNU/Linux distribution in order to tell users that they were getting a 'bonus'. Others then concluded that they had to compete with that, and* *began to offer non-free 'bonuses' too. The result is that today we have tens of millions of people using a version of the GNU system, and yet the goal of making a completely free operating system has been not just forgotten but almost totally cancelled." Would you agree with this?*

ER: There is not one sentence in that paragraph that I agree with! When Richard says the goal of making a completely free operating system has been forgotten, he's just posturing for effect. I can assure you that Linus Torvalds hasn't forgotten; nor have I; nor have any of the other million or so dedicated hackers out there. That's still where we're going, whether or not Richard chooses to acknowledge it.

RP: *Let's take another example: Stallman says that most GNU/Linux distributions now include non-free drivers. Does it concern you that non-Free Software is being shipped with Free Software?*

ER: It's true that there are distributions of Open Source operating systems that include binary data for some of the drivers, and that this software cannot be classified as either Open Source or Free Software. But while I am concerned about it, for me it is a practical issue, not a moral one. Personally, I don't use such drivers because I buy hardware that doesn't require using them.

RP: *Stallman is not the only person concerned that the pragmatic approach introduces risks. Jesus Villasante, the head of software technologies at the European Commission's Information Society and Media Directorate General, for instance, has argued that Open Source developers are being exploited by large corporations, who are using them as subcontractors, rather than encouraging the community to develop independent commercial products.*[56] *Does not the Open Source Movement risk being co-opted by the traditional software industry and emasculated in the process?*

ER: [laughs] I am laughing because it is fine if these companies think they are co-opting us. In reality, however, it is we who are co-opting them. We are changing their culture, their products, and their way of doing things much more than they are changing us. That is evident simply in the amount of Open Source code that is released every year.

Goodbye GPL?

RP: *You have said that you no longer believe there is any need for the GPL.*[57] *What's your thinking here?*

[56] "Companies are using the potential of communities as subcontractors — the open-source community today (is a) subcontractor of American multinationals," Villasante said at a debate on open-source innovation at the Holland Open Software Conference in Amsterdam in May 2005. Consequently, he called on the open-source community to develop more independence from these large companies. http://news.zdnet.com/2100-3513_22-5726714.html

[57] Written by Stallman in 1988, the General Public Licence is the first and most frequently used Free Software licence. It is currently being updated from Version 2 to Version 3, a process that is proving challenging and somewhat controversial. The new licence is expected to be completed by 2007. This interview was undertaken before this process began. Consequently there is no discussion of it with Raymond.

ER: My reasoning is that you can believe two things about Open Source: you can believe it is a superior system of production, in the sense that it is economically more efficient; or you can believe it is necessary to punish defectors. It is my belief that if Open Source really is a superior system of production, then doing what the GPL tries to prevent is its own punishment. It is for that reason that I no longer think that the GPL is necessary.

***RP:** What you are questioning is the need for the "viral", or reciprocal, component of the GPL[58]: the requirement that anyone who uses code licensed under the GPL must license any modifications to that code under the same free conditions. In saying this is unnecessary, you are assuming that no one would want to take the code from an Open Source product and create a proprietary version because they could never match the quality of the Open Source version?*

ER: That's right.

***RP:** I'm told that you personally favour the BSD licence?[59]*

ER: Right. I use the BSD for pretty much everything I do now. I could imagine a better licence, but there's no better licence that's really established.

***RP:** The BSD is not a reciprocal licence. In effect, then, developers can already take Open Source code and use it to create proprietary products. Unlike Stallman, then, you do not object to proprietary software per se?*

ER: No, I also don’t think it is morally wrong. However, I do think it's a dumb way to do things.

***RP:** What are your views on the intellectual property system in general?*

ER: I am part of a relatively small minority in the hacker community that actually favours strong intellectual property protection. I believe that creators should have control over the dissemination of their works; and they should be able to retain the right to keep it proprietary and sell various pieces of their bundle of rights on the market for whatever they can get.

That said, the IP system we have now is horribly broken. The term on copyright, for instance, is far too long; and the term on patents is arguably too long. In fact, the patent system is totally screwed up, as there is no obviousness bar on patents. That is wrong and needs to be fixed.

[58] Stallman objects strongly to the use of the term viral. After reading a 2003 article I wrote he e-mailed me to say: "To compare anything to a virus is extremely unfriendly. As regards the GPL, it is also inaccurate. The GPL's domain does not spread by proximity or contact, only by deliberate inclusion of GPL-covered code in your program. It spreads like a spider plant, not like a virus. People who hate the GPL have the right to say that it "contaminates" other software; that's misleading, but they have freedom of speech. However, if you don't hate the GPL, would you please not use smear words like "viral" and "contaminate" to describe it?"

[59] The Berkeley Software Distribution licence agreement. http://www.opensource.org/licenses/bsd-license.php

So my support for intellectual property rights doesn't really translate into supporting the existing system, which I feel is pretty much a disaster — at least in the United States.

***RP:** So you have no objection in principle to software and business method patents?*

ER: I don't. However, the problem is that patent examiners — at least in the US — lack competence, and are overworked and understaffed. In addition, the incentives built into their bureaucracy, and into their pay structure, mean that they are more inclined to issue patents than they are to deny them.

***RP:** What are your views on "trusted computing"?*[60]

ER: On this Stallman is right. He calls it "treacherous computing," and that is a totally appropriate way to describe it. In reality, it is a way to take control of the hardware out of the hands of the user.

***RP:** For Free and Open Source developers there is also the threat that their software may never work on "trusted" hardware.*

ER: Exactly. What trusted computing means is that Microsoft can guarantee that Linux will never boot on another Intel chip [laugh].

***RP:** Given the heated debate surrounding trusted computing, I guess it may never happen.*

ER: Right. I don't think it will. Pressure for trusted computing is coming from two places: Microsoft wants to use it as a tool to increase its software monopoly; and the big media companies — through the MPAA[61] and RIAA[62] — want to use it as a tool for DRM.[63] But if you look at the statistics you will see that the hardware, computer and consumer electronics industries are six times the size of the entertainment

[60] Trusted computing is a family of open specifications whose stated goal is to make personal computers more secure through the use of dedicated hardware. Critics, including academics, security experts, and users of free and Open Source software, contend, however, that the overall effect (and perhaps intent) of trusted computing is to impose unreasonable restrictions on how people can use their computers.

[61] The Motion Picture Association of America (MPAA), originally called the Motion Pictures Producers and Distributors Association, is a non-profit trade association based in the United States which was formed to advance the interests of movie studios. Its members consist of the "big seven" major Hollywood studios. The organisation produces the well-known voluntary film rating system.

[62] The RIAA was formed in 1952. Its primary purpose at that time was to administer the RIAA equalisation curve. This is a technical standard of frequency response applied to vinyl records during manufacturing and playback. The RIAA has continued to participate in creating and administering technical standards for later systems of music recording and reproduction, including magnetic tape, cassette tapes, digital audiotapes, CDs and software-based digital technologies. The RIAA also participates in the collection, administration and distribution of music licenses and royalties. The RIAA has been at the heart of the file-sharing controversy, especially music files in the popular MP3 format uploaded onto the Internet using peer-to-peer software.

[63] Digital rights management (DRM) is the umbrella term referring to any of several technologies used to enforce pre-defined policies controlling access to software, music, movies, or other digital data. DRM critics argue that the phrase "digital rights management" is a misnomer and the term "digital restrictions management" is a more accurate characterisation of the functionality of DRM systems. http://en.wikipedia.org/wiki/Digital_rights_management

industry. From the point of view of society, therefore, DRM is a bad economic trade-off. Moreover, since Sony earns six times more revenue from electronic hardware it seems clear which side it will eventually come down on.

RP: *I note you are using trusted computing synonymously with DRM?*

ER: Right. There are technical differences between them, but trusted computing is essentially a tool for DRM.

Not magic pixie dust

RP: *I'd like to raise with you a number of points that people have made in connection with Open Source and your views on it. Nickolai Bezroukov,*[64] *for instance, has argued that having access to the source code in complex software projects is unlikely to give other programmers sufficient understanding of how a program works. What is also needed, he said, is knowledge about the underlying architecture. In short, access to the code alone is not enough.*

ER: Like a lot of commentators, Bezroukov misses an important secondary phenomenon. That is that the point of peer review is not just that people practice it, but that its existence changes behaviour. When I know that someone is going to look at my code, for instance, I write it in a different way. I am more careful about my assumptions and careful to make my code readable. Process transparency has important effects even if nobody actually looks at the code.

RP: *One of the central arguments in The Cathedral & the Bazaar is that good software is produced by letting programmers "scratch their itch". Another of your critics, ZDNet columnist John Carroll, has challenged this claim, arguing that while it may lead to the production of software that programmers want, it doesn't usually produce software that customers want.*[65]

ER: Whether the software developed is good for end users depends on what kind of itch the programmer has! Some have an itch to develop programs that are good for Aunt Tillie; some have an itch to produce programs that are good for other programmers.

RP: *You have yourself acknowledged that applications software is not an area where Open Source has been successful.*

ER: Actually, we are not bad at applications design: it is end-user interfaces where we tend to fall down. There are two mitigating factors here: one is that, with the one exception of the Macintosh people, everybody else sucks at it too — the Windows crowd certainly isn't any better at it than we are [laughs].

[64] *Open Source Software Development as a Special Type of Academic Research (Critique of Vulgar Raymondism*) Nikolai Bezroukov, *First Monday*, vol 4 no 10, October 1999 http://www.firstmonday.dk/issues/issue4_10/bezroukov/index.html

[65] *Proprietary drives the software economy*, John Carroll, ZDNet, May 10, 2004 http://news.zdnet.com/2100-9595_22-5209078.html

The second point is that we are — slowly and painfully — getting better. KDE and GNOME[66] today are by no means perfect, but they represent a huge advance on where we were in 1997.

***RP:** Let's return for a moment to the Netscape decision to release the source code of its browser. Without doubt this was a huge PR coup for you, but it was followed by a period of considerable stagnation, confusion, and disappointment. Eventually Jamie Zawinski, one of the lead developers,[67] resigned — on the grounds that the project had completely failed to deliver a usable product. It's true that the Firefox browser[68] eventually emerged from the project, but only after a lot of code had to be rewritten. Open Source is not a cure for all ills, is it?*

ER: Jamie was right when he said that Open Source is not magic pixie dust.[69] As Fred Brooks observed in the 1970s, there are no silver bullets in software design. But the point to bear in mind is that proprietary software can no longer handle complexity beyond a certain level. It just ran out of steam.

***RP:** So whatever its faults, Open Source is still better than closed source?*

ER: Right. A proprietary model was necessary initially because computing resources, and the high cost of the machines themselves, were high. Since software complexity was low, this did not matter. Today, however, complexity has reached a level where closed source simply cannot cope, and Open Source peer review has become essential. Fortunately, as complexity has gone up, computing costs have come down.

But let's be clear: I am in no doubt that in the future we will reach a complexity scale where Open Source peer review will itself be insufficient to keep error rates reasonable. That does not mean that we will have to abandon Open Source, but simply that we will have to come up with some additional ways to control and manage complexity: maybe some development of formal proofs of correctness or automated code generation. I don't know.

***RP:** What do you think has been achieved by the Open Source Movement, and what has still to be done?*

ER: The most important thing we have achieved is to win the quality argument. You can't find anybody who is not on Microsoft's payroll who still maintains that Open Source is inferior. So the principle that peer review and transparency leads to better software has soaked into the background culture and we don't need to argue it anymore.

[66] The two primary Open Source desktop environments http://en.wikipedia.org/wiki/GNOME and http://en.wikipedia.org/wiki/Kde

[67] Jamie W. Zawinski, commonly known as jwz, is a computer programmer, responsible for significant contributions to the free software projects Mozilla and XEmacs. He is also the proprietor of the DNA Lounge, a nightclub in San Francisco. http://en.wikipedia.org/wiki/Jamie_Zawinski. He resigned from Netscape in April 1999.

[68] http://www.mozilla.com/firefox/

[69] Zawinski ended his resignation letter saying: "If there's a cautionary tale here, it is that you can't take a dying project, sprinkle it with the magic pixie dust of "open source," and have everything magically work out. Software is hard. The issues aren't that simple."

Second, we have become entrenched. We now have a huge degree of corporate support, and companies like IBM and Novel can announce support for Linux and no one even thinks it weird any more. That is a hell of a contrast compared to 1999.

That said, there remain huge issues: we could still get badly spiked by, for example, the consequences of an aggressive patent blitz, or draconian application of the DMCA.[70] So our war isn't yet over.

***RP:** What do you see as the end game of the Open Source Movement?*

ER: From my point of view the goal is a world in which — barring the odd sporadic exception — all software is open, all protocols are open, all file formats are documented, and users have complete control over their computers. Oh, and the most important thing: programmers don't have to sit bolt upright in their bed at 3 o'clock in the morning wondering: "Why do I put in so much work and effort, and it all sucks so bad."

***RP:** As you say, Open Source is now entrenched. There are also a growing number of other "open" movements, including Open Access, Open Spectrum, Open Biology, Open Politics, Creative Commons, and so on. What common threads do you see between all these movements?*

ER: These other groups are all very obviously influenced by the Open Source Movement.

***RP:** So is it just that the Open Source Movement created a compelling model that others are trying to apply elsewhere, or is there some driving force out there demanding greater openness in many areas?*

ER: That's a good question, and it has a deep answer — and the deep answer is that centralisation doesn't scale.

***RP:** Why is that an issue today, rather than, say, fifty years ago?*

ER: Because the population is scaling up, the size and complexity of our economy is scaling up, and the number of bits of information that human beings have to deal with every day in order to function is scaling up.

One of the things we learn from history is that, as economies scale up, centralised systems become less and less appropriate and effective. And it turns out this doesn't just affect centrally planned economies: it applies to any kind of command-and-control system, anywhere.

[70] Signed into law by President Bill Clinton on October 28, 1998, the DMCA was passed in order to implement the 1996 World Intellectual Property Organisation (WIPO) Copyright Treaty. The Bill included controversial anti-circumvention prohibitions that criminalise production and dissemination of technology that can circumvent measures taken to protect copyright, not merely infringement of copyright itself. Essentially, these provisions were designed to stop people circumventing digital rights management systems.

RP: *And greater complexity demands greater transparency and openness?*

ER: Yes.

Market anarchist

RP: *Let's move on to your politics. In his article Bezroukov accused you of vulgar Marxism. You took exception to that?*

ER: Yes, that really pissed me off.

RP: *Clearly you are not a Marxist, but how would you describe your politics?*

ER: I'm a market anarchist.

RP: *You also describe yourself as a libertarian.*

ER: I do. I should explain that in the US, a libertarian means that you are either a minimum government Jeffersonian (which is about 75% of libertarians) or an out-and-out anarchist (which is the other 25%).

RP: *What do you mean when you say you are a market anarchist?*

ER: It means I want to live in a society that is legally ordered but without a government; with no one having a monopoly on using force.

RP: *Governments, of course, are the ultimate command-and-control systems. You, in fact, believe that the Internet will one day make them obsolescent. Indeed, you're not alone in believing that the Web will change the way we govern ourselves. There is, for instance, a view that it will allow "emergent democracy"*[71] *to develop?*

ER: Right, but democracy isn't very interesting to me. It is only relevant as a transition stage: as a way of limiting the power of the state. I do not think it is anything like the final answer. What I would want to see is emergent anarchy.

RP: *Presumably in an anarchy there would be no leaders?*

ER: In a functioning anarchy you can have influence leaders who are followed voluntarily because they are respected; but you can't have people who have the ability to put a gun to somebody's head and say jump. So you can have people who have influence, but you can't have people who give orders.

One thing I should stress here is that, while I am willing to talk about my politics, I do so on the understanding that I don't think anyone needs to buy my politics to buy Open Source.

[71] See, for instance, *Weblogs and Emergent Democracy* by Joi Ito at http://joi.ito.com/static/emergentdemocracy.html. Joi Ito is a Japanese-born, American-educated, activist, entrepreneur, and venture capitalist. http://en.wikipedia.org/wiki/Joi_Ito

RP: *Sure. But do you personally see connections between Open Source and your politics?*

ER: I do.

RP: *It strikes me, for instance, that the maintainers of Open Source projects operate as "influence leaders"?*

ER: Right.

RP: *When you wrote Homesteading the Noosphere,*[72] *one of your objectives was to explain how thousands of hackers working together in an apparently chaotic collaborative manner are able to establish and maintain a high degree of order and control.*

ER: Yes. I wanted to show the way in which the implicit norms that have developed in the Movement are very similar to common law land tenure, both of which are quasi-instinctive, or wired in the system of property rights, that human beings develop anywhere where they have rights to commodities that are worth defending.

RP: *These same implicit norms would be harnessed in an anarchist community?*

ER: That's right. There is a kind of spontaneous social order that arises from people's selfish desire to minimise violence amongst themselves. This is not something that needs to be imposed. It is something that gets reinvented over and over. I recently read, for example, a study on social coercion and violence in the Virginia mining camps in the mid 1800s — during the Nevada silver rush. Government authority hadn't gotten there yet, so there was no law enforcement; and yet the miners developed an extremely effective system of dispute mediation and violence suppression. Basically nobody wanted to get shot!

RP: *This affinity between your politics and your interest in the Open Source Movement is interesting, but what came first: the politics or your anthropologising of hackers?*

ER: Well, my politics arrived at essentially the form I hold them in today in the early 1980s, which was many years before I anthropologised the Open Source Movement; so I would have to say that my politics were formed before I found myself in a leadership position in the Open Source Movement.

Having said that, I think there is one important causal connection, and that is my background as a libertarian, and my sensitivity to scaling phenomena and the failure of our current political system. I think this was significant in that it enabled me to see social patterns that other people might miss — because they are constantly looking for a central point, a leader. I, on the other hand, had been thinking for many years about

[72] *Homesteading the Noosphere*, Eric Raymond, published in *The Cathedral & the Bazaar*, O'Reilly, 1999

societies, and the way in which pure networks — with no central point of authority — work. Consequently, I think I was prepared to see things that others didn't.

***RP:** While you are anti-government you are clearly not anti-market. However, I'm struck that in your essay Revenge of the Hackers,[73] you described the Movement as "the revenge of the hackers after 20 years of marginalisation." When it comes to hackers and businesses, we're talking about a cultural war, aren't we?*

ER: Oh, I have never denied that that aspect exists! [laughs].

***RP:** Indeed, hacker culture is inherently antithetical to corporate culture. How can these two worlds ever fruitfully co-exist?*

ER: Well, while it is correct to say that the hacker culture is anti-authoritarian that does not mean that it is, therefore, anti-market.

***RP:** But the only world that corporations understand is an authoritarian one, isn't it?*

ER: [pause] Um, I'm not sure I would completely agree with that because corporations are not completely authoritarian organisations. You don't get shot if you try to quit, for instance! [laughs].

***RP:** Maybe, but they are very hierarchical.*

ER: They are, but they are not authoritarian in the way the military is during a war.

***RP:** It also occurs to me that while openness is — by definition — central to Open Source philosophy; it is not a concept that corporations are generally comfortable with.*

ER: True, so the whole point of my efficiency argument is that virtue is its own reward. Companies can actually get an economic reward from abandoning secrecy and centralised command-and-control.

***RP:** You said earlier that the Open Source Movement was more likely to co-opt business than vice versa. Some might feel that the Open Source Movement is a potential Trojan Horse, capable of infiltrating the corporate world and subverting it. Is that far fetched?*

ER: No, actually it isn't. It is something I don't talk about a lot, but it is indeed part of my thinking.

***RP:** So your thinking is that corporate culture can be subverted by the hacker ethos. We're talking about a kind of entryism?[74]*

ER: Yes. And I say that to business people too! I don't believe in hidden agendas. The most effective way to change people's behaviour is to use manipulation

[73] *Revenge of the Hackers*, Eric Raymond, published in *The Cathedral & the Bazaar*, O'Reilly, 1999

[74] Entryism is a political tactic by which an organisation encourages members to infiltrate another organisation in an attempt to gain recruits, or take over entirely.

techniques that work even when the person knows you are using them. Not only are those the most effective way to change people's behaviour, but they are the only ones that I consider ethical.

***RP:** So how would you respond if you were asked if your support for Open Source was part of a wider political agenda?*

ER: Well, in so far as Open Source is a demonstration of certain things it is. And I hope that that demonstration will have long-term political consequences.

***RP:** So there is a definite connection between Open Source and your politics?*

ER: As I said, I think my politics prepared me to see some important things; and, as I also said, I wouldn't be sorry if Open Source did turn out to be a form of entryism that taught people a different model for how to cooperate than they have right now.

At the same time I am very sincere when I say that I don't need people to buy my politics to do the Open Source thing, and I don't want people to think my politics are a barrier. Open Source has to stand or fall by itself, not on any crazy or half-baked political thoughts I might have.

Guns and Myths

***RP:** You said that you don't trust governments. You are also a firm supporter of Second Amendment rights. Do you believe that citizens need guns in order to protect themselves from the government?*

ER: Absolutely. This is something as a European you will have difficulty understanding. Although I am more upfront and more articulate about this than most, it is a very widespread point of view amongst Americans. Love your country, but never, *ever* trust your government.

***RP:** Is not the concept of an armed citizenry anachronistic in the modern world? Faced with insurgency, a government could just drop something nasty from the sky?*

ER: I don't think so. The situation would have to have degenerated pretty far before a tyrant would be prepared to bomb his own citizens, and citizen arms are often effective in changing the balance of power before that point. Ask the Lithuanians who staged a citizen's coup with citizen's firearms and overthrew their government. Ask the Russians who came out in support of perestroika when that was threatened.

***RP:** I think most people view guns more as a deterrent against burglars and violent criminals, rather than their government.*

ER: Well, to me the two cases aren't really ethically distinguishable. I don't draw much distinction between criminals and governments — both groups engage in violence that is not defensive.

***RP:** Do you sleep with a gun under your pillow, just in case?*

ER: I won't comment on how I keep it stored because people not knowing is part of my security.

***RP:** You said earlier that you are a neo-pagan. You are also a Wiccan and once led a coven.[75] How would you explain neo-paganism to a layperson?*

ER: Oh boy, that one is as complicated as everything else we have talked about together. It means that I am associated with a group of religions that enable you to experience things that you wouldn't ordinarily, and gain capabilities that you wouldn't ordinarily. And, unlike conventional religions, they judge you primarily by how competent you are at those specialist skills, not by how strongly you believe.

***RP:** Is this spiritual, or are we talking about a states-of-consciousness thing?*

ER: I don't recognise the distinction. Your question assumes a distinction I don't believe exists. "Spirituality" is all about non-ordinary states of consciousness.

***RP:** Does it involve the supernatural?*

ER: Some neo-pagans are supernaturalist, but most are not. Those who are not supernaturalist believe it to be a nonsense concept: either the phenomena you observe are causally connected to the rest of the universe, in which case they can be understood in the same way the rest of the universe is; or they are not causally connected to the rest of the universe, in which case they are irrelevant.

***RP:** Commenting on your web site about the Sam Williams'[76] biography of Richard Stallman, you said that Williams had failed to challenge the myths Stallman has created about himself. You too create myths about yourself, don't you?*

ER: Well, I try not to say anything that isn't true [laughs]. While I am not accusing Richard of lying about himself, he is selective in the way he emphasises elements of his own past. I don't know, maybe I do that too: it would take a third party looking dispassionately at the historical record of both of us to evaluate that.

***RP:** I'm thinking of some comments you made in The Hacker's Revenge. There you talked about the need for the spokesperson of the Open Source Movement to project an image of "attractive dissonance". The aim, you said, should be to excite the interest of journalists. In effect, you were talking about branding yourself, of creating some myths about Eric Raymond.*

[75] Wicca is a Neopagan religion found in many different countries, though most commonly in English-speaking cultures. Wicca was first publicised in 1954 by a British civil servant and Co-Freemason named Gerald Gardner after the British Witchcraft Act was repealed. He claimed that the religion, of which he was an initiate, was a modern survival of an old witch cult, which had existed in secret for hundreds of years, originating in the pre-Christian Paganism of Europe. Wicca is thus sometimes referred to as the Old Religion. http://en.wikipedia.org/wiki/Wiccan

[76] *Free as in Freedom*, Sam Williams, O'Reilly, 200x

ER: Well, you get into an interesting definitional thing here. Where do you draw the line between myth-making and normal self-presentation? I ask that not as a reductive question; it's a serious question I grapple with all the time.

My public persona is me, there's nothing false about it. But it's a sort of exaggerated me, with the parts I think are important for the public message pumped so far up that some of the other bits of me don't show.

***RP:** Stallman certainly accuses you of creating myths about yourself. He said to me for instance: "Eric Raymond paints himself as being closely involved and a former supporter of the GNU Project. But he never was." How would you respond to that?*

ER: Richard doesn't deny that I was one of the first people he told about the project; nor that I suggested that Emacs be its first product; nor that my wife Cathy (an attorney) and I worked together in reviewing early drafts of GPL 2.0. In fact, I am *still* a supporter of the GNU Project. It's code, anyway, if not its moral theories and propaganda. I have a twenty-year history of doing necessary scutwork[77] that the FSF's own coders won't touch.

***RP:** I did ask Stallman to clarify whether he was saying that you had never written code for the GNU Project. He replied: "He has written Free Software, and he contributed some to GNU Emacs, but he was not one of the early major contributors. In general, he tries to exaggerate his involvement. It fits the mental pattern of the supporter who now rejects the activity because he thinks it is wrong. I think he does that so that when he criticises us his criticisms will have more weight."*

ER: There are quite a number of things Richard seems to have forgotten. But I'm not really interested in arguing about whether I count as a "major contributor" or not. Your readers are invited to grep[78] for submitter names in the LISP libraries and draw their own conclusions.

I have never allowed Richard's tendency to feud to prevent me from offering contributions to Emacs or the GNU Project in general, and don't intend to let it in the future either. Because something I understand, even if Richard sometimes forgets it, is that all the politicking and arguments about priority are ultimately bullshit. It matters far less to me what Richard thinks about my support of GNU than it does to him. That's why I don't care to argue about whether I'm a "major contributor" or not. That's a definitional, political dispute that doesn't matter. The code is what matters.

[77] Scutwork means monotonous work or menial tasks that have to be done, usually as part of a large complex job or project. http://www.thefreedictionary.com/scutwork

[78] grep is a command line utility originally written for use with the UNIX operating system. The default behaviour of grep takes a regular expression on the command line, reads standard input or a list of files, and outputs the lines containing matches for the regular expression. The name comes from a command in the UNIX text editor ed that takes the form: g/re/p which means "search globally for lines matching the regular expression, and print them. http://en.wikipedia.org/wiki/Grep

RP: *You have a reputation for being somewhat pugnacious. John Carroll, for instance, described you as the kind of person who in a café debate might be likely to "take out his opponent's eye with a fork."*[79] *Would you agree?*

ER: Wow. Do I strike you as someone who would want to take your eye out with a fork?

RP: *You've been completely civil to me.*

ER: Well, I can say that my interaction with you is far more typical than my interaction with John Carroll. Maybe he and I just had bad chemistry. I don't usually see any need to take people's eyes out with a fork because that kind of attitude really stems from lack of confidence — which I don't normally suffer from.

RP: *There have been moments. In 1999, for instance, you sent a threatening e-mail to Bruce Perens in which you said: "If you ever again behave like that kind of disruptive asshole in public, insult me, and jeopardise the interests of our entire tribe, I'll take it just as personally — and I will find a way to make you regret it. Watch your step." Perens posted the message to a mailing list and indicated that as you were a gun enthusiast he had reported the incident to the police.*

ER: That was pure grandstanding on his part, and I have never forgiven him for it. But let's be clear about this: I wasn't angry with him for disagreeing with OSI's position, but for criticising me in public rather than coming to OSI privately to express his concerns.[80]

RP: *A 1998 article in the Washington Post said that your wife supported you. Are you still financially reliant on her?*

ER: That's hard to say because the pattern is that she covers the day-to-day expenses and every once in a while I give in a large lump sum as my contribution to household funds. So over time it is hard to say whether she supports me or not.

RP: *At one point you had the potential to provide a very sizeable lump sum: as a board member of VA Linux during its 1999 IPO,*[81] *you were given 150,000 shares. On the day the company floated, these were worth $52 million. Do you still own the shares?*

ER: I still have them.

[79] *How the software economy is driven by proprietary work*, John Carroll, ZDNet , May 21st 2004 http://www.zdnet.co.uk/print/?TYPE=story&AT=39154927-39020682t-21000005c

[80] The dispute took place when Perens publicly criticised Raymond for giving his blessing to the Apple Public Source License (APSL), which Perens maintained did not meet the requirements of Open Source software. http://news.com.com/2100-1001-223388.html

[81] VA Software is notable because of its Initial Public Offering on December 9, 1999. The shares were offered at $30, but opened at $299, and went up to $320 before closing their first day of trading at $239.25, a 698% return — this earned the company a reputation for being the most "successful" IPO of all time. However, this high-flying success was short-lived, and within a year the stock was selling at well below the initial offer price, in a classic example of the dot-com stock market bubble. Many authors of free software were invited to buy shares at the initial price offering as part of a friends and family deal. http://en.wikipedia.org/wiki/Va_linux

RP: *Last time I looked that $52 million had fallen away to $250,000. Do you wish you had cashed them in and made a killing?*

ER: [laughs] It would have been nice, but I was worrying about other things at the time! The thing is I don't care about money that much. I am more interested in changing the world.

RP: *You also believe in free love?*

ER: Hah!

RP: *How does that work?*

ER: I will say that I hang around with polyamorists[82] and [pause] … um… to say more would probably be too distracting.

RP: *Are you declining to answer the question?*

ER: I'm answering by saying that I hang around with polyamorists.

RP: *OK. Let's leave it at that then. Thank you very much for taking time to speak with me.*

If you would like to republish the article on a commercial basis, or have any comments on it, please email me at richard.poynder@journalist.co.uk.

Please note that while I make this interview freely available to all, I am a freelance journalist by profession, and so make my living from writing. To assist me to continue making my work available in this way I invite anyone who reads this interview to make a voluntary contribution. I have in mind a figure of $8, but whatever anyone felt inspired to contribute would be fine. This can be done quite simply by sending a payment to my PayPal account, quoting the email address *aotg20@dsl.pipex.com*. It is not necessary to have a PayPal account to make a payment.

I would like to acknowledge the help of the Open Society Institute, which provided a small upfront grant to enable me to get started on *The Basement Interviews* Project. Further information about *The Basement Interviews* can be found at the Open and Shut? site.

A companion interview with Richard Stallman is available here.

[82] Polyamory, in its broadest usage, is the practice or lifestyle of being open to having more than one loving, intimate relationship at a time, with the full knowledge and consent of all partners involved. Persons who consider themselves emotionally suited to such relationships may define themselves as polyamorous or polysexual (a blanket term), often abbreviated to poly. http://en.wikipedia.org/wiki/Polyamorists