This article has originally been published in German as part of the

available at www.opensourcejahrbuch.de.

The *Open Source Jahrbuch 2007* is an extensive compendium dealing with the various aspects of open source software and beyond. Whilst most articles have been written in German, this is one of the articles that have originally been written in English and subsequently been translated into German. Refer to our website for more English articles.

Why "Open Source" Misses the Point of Free Software

FROM THE FREE SOFTWARE MOVEMENT: RICHARD STALLMAN

1 Introduction

When we call software "free," we mean that it respects the users' essential freedoms: the freedom to run it, to study and change it, and to redistribute copies with or without changes. This is a matter of freedom, not price, so think of "free speech," not "free beer."

These freedoms are vitally important. They are essential, not just for the individual users' sake, but because they promote social solidarity—that is, sharing and cooperation. They become even more important as more and more of our culture and life activities are digitized. In a world of digital sounds, images and words, free software comes increasingly to equate with freedom in general.

Tens of millions of people around the world now use free software; the schools of regions of India and Spain now teach all students to use the free GNU/Linux operating system. But most of these users have never heard of the ethical reasons for which we developed this system and built the free software community, because today this system and community are more often described as "open source," and attributed to a different philosophy in which these freedoms are hardly mentioned.

The free software movement has campaigned for computer users' freedom since 1983. In 1984 we launched the development of the free operating system *GNU*, so we could avoid the non-free operating systems that deny freedom to their users. During the 80s, we developed most of the essential components of such a system, as well as the *GNU General Public License*, a license designed specifically to protect freedom for all users of a program.

However, not all of the users and developers of free software agreed with the goals of the free software movement. In 1998, a part of the free software community splintered off and began campaigning in the name of "open source." The term was originally proposed to avoid a possible misunderstanding of the term "free software,"

but it soon became associated with philosophical views quite different from those of the free software movement.

Some of the proponents of "open source" considered it a "marketing campaign for free software," which would appeal to business executives by citing practical benefits, while avoiding ideas of right and wrong that they might not like to hear. Other proponents flatly rejected the free software movement's ethical and social values. Whichever their views, when campaigning for "open source" they did not cite or advocate those values. The term "open source" quickly became associated with the practice of citing only practical values, such as making powerful, reliable software. Most of the supporters of "open source" have come to it since then, and that practice is what they take it to mean.

Nearly all open source software is free software; the two terms describe almost the same category of software. But they stand for views based on fundamentally different values. Open source is a development methodology; free software is a social movement. For the free software movement, free software is an ethical imperative, because only free software respects the users' freedom. By contrast, the philosophy of open source considers issues in terms of how to make software "better"—in a practical sense only. It says that non-free software is a suboptimal solution. For the free software movement, however, non-free software is a social problem, and moving to free software is the solution.

Free software. Open source. If it's the same software, does it matter which name you use? Yes, because different words convey different ideas. While a free program by any other name would give you the same freedom today, establishing freedom in a lasting way depends above all on teaching people to value freedom. If you want to help do this, it is essential to speak about "free software."

We in the free software movement don't think of the open source camp as an enemy; the enemy is proprietary software. But we want people to know we stand for freedom, so we do not accept being misidentified as open source supporters.

2 Common Misunderstandings of "Free Software" and "Open Source"

The term "free software" has a problem of misinterpretation: an unintended meaning, "Software you can get for zero price," fits the term just as well as the intended meaning, "software which gives the user certain freedoms." We address this problem by publishing the definition of free software, and by saying "Think of free speech, not free beer." This is not a perfect solution; it cannot completely eliminate the problem. An unambiguous, correct term would be better, if it didn't have other problems.

Unfortunately, all the alternatives in English have problems of their own. We've looked at many alternatives that people have suggested, but none is so clearly "right" that switching to it would be a good idea. Every proposed replacement for "free

software" has some kind of semantic problem—and this includes "open source software."

The official definition of "open source software" (which is published by the *Open Source Initiative* and too long to cite here) was derived indirectly from our criteria for free software. It is not the same; it is a little looser in some respects, so open source supporters have accepted a few licenses that we consider unacceptably restrictive of the users. Nonetheless, it is fairly close to our definition in practice.

However, the obvious meaning for the expression "open source software" is "You can look at the source code," and most people seem to think that's what it means. That is a much weaker criterion than free software, and much weaker than the official definition of open source. It includes many programs that are neither free nor open source.

Since that obvious meaning for "open source" is not the meaning that its advocates intend, the result is that most people misunderstand the term. Here is how writer Neal Stephenson defined "open source":

> "Linux is 'open source' software meaning, simply, that anyone can get copies of its source code files."

I don't think he deliberately sought to reject or dispute the "official" definition. I think he simply applied the conventions of the English language to come up with a meaning for the term. The state of Kansas published a similar definition:

> "Make use of open-source software (OSS). OSS is software for which the source code is freely and publicly available, though the specific licensing agreements vary as to what one is allowed to do with that code."

The open source people try to deal with this by pointing to their official definition, but that corrective approach is less effective for them than it is for us. The term "free software" has two natural meanings, one of which is the intended meaning, so a person who has grasped the idea of "free speech, not free beer" will not get it wrong again. But "open source" has only one natural meaning, which is different from the meaning its supporters intend. So there is no succinct way to explain and justify the official definition of "open source." That makes for worse confusion.

3 Different Values Can Lead to Similar Conclusions . . . But Not Always

Radical groups in the 1960s had a reputation for factionalism: some organizations split because of disagreements on details of strategy, and the two daughter groups treated each other as enemies despite having similar basic goals and values. The right-wing made much of this, and used it to criticize the entire left.

Some try to disparage the free software movement by comparing our disagreement with open source to the disagreements of those radical groups. They have it backwards. We disagree with the open source camp on the basic goals and values, but their views and ours lead in many cases to the same practical behavior—such as developing free software.

As a result, people from the free software movement and the open source camp often work together on practical projects such as software development. It is remarkable that such different philosophical views can so often motivate different people to participate in the same projects. Nonetheless, these views are very different, and there are situations where they lead to very different actions.

The idea of open source is that allowing users to change and redistribute the software will make it more powerful and reliable. But this is not guaranteed. Developers of proprietary software are not necessarily incompetent. Sometimes they produce a program which is powerful and reliable, even though it does not respect the users' freedom. How will free software activists and open source enthusiasts react to that?

A pure open source enthusiast, one that is not at all influenced by the ideals of free software, will say, "I am surprised you were able to make the program work so well without using our development model, but you did. How can I get a copy?" This attitude will reward schemes that take away our freedom, leading to its loss.

The free software activist will say: "Your program is very attractive, but not at the price of my freedom. So I have to do without it. Instead I will support a project to develop a free replacement." If we value our freedom, we can act to maintain and defend it.

4 Powerful, Reliable Software Can Be Bad

The idea that we want software to be powerful and reliable comes from the supposition that software is meant to serve its users. If it is powerful and reliable, it serves them better. But software can only be said to serve its users if it respects their freedom. What if the software is designed to put chains on its users? Then reliability only means the chains are harder to remove.

Under the pressure of the movie and record companies, software for individuals to use is increasingly designed specifically to restrict them. This malicious feature is known as *DRM*, or *Digital Restrictions Management*[1], and it is the antithesis in spirit of the freedom that free software aims to provide. And not just in spirt: since the goal of *DRM* is to trample your freedom, *DRM* developers try to make it hard, impossible, or even illegal for you to change the software that implements the *DRM*.

Yet some open source supporters have proposed "open source DRM" software. Their idea is that by publishing the source code of programs designed to restrict your access to encrypted media, and allowing others to change it, they will produce more

1 See http://www.DefectiveByDesign.org.

powerful and reliable software for restricting users like you. Then it will be delivered to you in devices that do not allow you to change it.

This software might be "open source," and use the open source development model; but it won't be free software, since it won't respect the freedom of the users that actually run it. If the open source development model succeeds in making this software more powerful and reliable for restricting you, that will make it even worse.

5 Fear of Freedom

The main initial motivation for the term "open source software" is that the ethical ideas of "free software" make some people uneasy. That's true: talking about freedom, about ethical issues, about responsibilities as well as convenience, is asking people to think about things they might prefer to ignore, such as whether their conduct is ethical. This can trigger discomfort, and some people may simply close their minds to it. It does not follow that we ought to stop talking about these things.

However, that is what the leaders of "open source" decided to do. They figured that by keeping quiet about ethics and freedom, and talking only about the immediate practical benefits of certain free software, they might be able to "sell" the software more effectively to certain users, especially business.

This approach has proved effective, in its own terms. The rhetoric of open source has convinced many businesses and individuals to use, and even develop, free software, which has extended our community—but only at the superficial, practical level. The philosophy of open source, with its purely practical values, impedes understanding of the deeper ideas of free software; it brings many people into our commnunity, but does not teach them to defend it. That is good, as far as it goes, but it is not enough to make freedom secure. Attracting users to free software takes them just part of the way to becoming defenders of their own freedom.

Sooner or later these users will be invited to switch back to proprietary software for some practical advantage. Countless companies seek to offer such temptation, some even offering copies gratis. Why would users decline? Only if they have learned to value the freedom free software gives them, to value freedom as such rather than the technical and practical convenience of specific free software. To spread this idea, we have to talk about freedom. A certain amount of the "keep quiet" approach to business can be useful for the community, but it is dangerous if it becomes so common that the love of freedom comes to seem like an eccentricity.

That dangerous situation is exactly what we have. Most people involved with free software say little about freedom—usually because they seek to be "more acceptable to business." Software distributors especially show this pattern. Nearly all GNU/Linux operating system distributions add proprietary packages to the basic free system, and they invite users to consider this an advantage, rather than a step backwards from freedom.

Proprietary add-on software and partially non-free GNU/Linux distributions find fertile ground because most of our community does not insist on freedom with its software. This is no coincidence. Most GNU/Linux users were introduced to the system by "open source" discussion which doesn't say that freedom is a goal. The practices that don't uphold freedom and the words that don't talk about freedom go hand in hand, each promoting the other. To overcome this tendency, we need more, not less, talk about freedom.

6 Conclusion

As the advocates of open source draw new users into our community, we free software activists have to work even more to bring the issue of freedom to those new users' attention. We have to say, "It's free software and it gives you freedom!"—more and louder than ever. Every time you say "free software" rather than "open source," you help our campaign.