Hybrid Innovation

The Dynamics of Collaboration
Between the FLOSS Community and Corporations

Yuwei Lin
Vrije Universiteit Amsterdam

Presentation at the FLOSS Symposium in Padova, Italy 14-15 April 2005

vrije Universiteit amsterdam

Cases of Hybrid Innovation

- Ubuntu GNU/Linux and Debian GNU/Linux

- StarOffice and OpenOffice.org (Sun)

- RedHat GNU/Linux and Fedora GNU/Linux (RedHat)

* OSS firms: Firms that develop, distribute, support FLOSS-based products or services

Outline of the Talk

- **A** Community of Open Source Practices (OSP)
- **W**orking Practices in OSS Firms
- **T**he Working Relationship Between the FLOSS Community and OSS Firms
- **Tensions** in the Hybrid Collaboration
- **T**he Heterogeneous FLOSS Social World

The FLOSS Innovation

The history of the IT industry certainly is a wonderful vision of the evolution of an autocatalytic knowledge set, catalysed by the actions of a diverse set of agents, driven by a diverse set of motives, all resulting in an explosion of economic activity and an avalanche of creation and destruction. The heart of this process of creative destruction is the epistemic cycle of uncertainty, imagination and innovation.

(Jackson, Mandeville & Potts 2002: 329)

The FLOSS Social World

- **A** Community of Open Source Practices (OSP)
- **C**elebrated Diversity
- **C**ross-Boundary
- **S**ocial Capitals Emerged from Networking
- **D**ifferent meanings are assigned by diverse actors to their collective practices (e.g. sharing source code) and common concepts (e.g. Free/Libre Open Source Software (FLOSS)).
- **M**ultiple Interpretations and Adjudication of Conflict Across Social Boundaries

Working Practices in OSS Firms

Routines

> *When I do the routine work I don’t always have to be at my desk. I get to decide when I get to do the things and also I have a lot of influences on what I do because we never get told ‘you have to do this’. It’s always we have these things that need to be done, who can do them? Then people can just volunteer. I mean, we always have plenty of things to do, so it’s just the question of picking the one that is interesting to you, which is more or less the same thing you do when you work in your free time. So there’s another freedom both in one I work and when I work home. But still then choices are of course made by my employer. I am not that just sponsor to sit and do whatever I want. There are some but not me.*
>
> (BO060203)

Working Practices in OSS Firms

Concerns on Time Pressure, Clients, and Finance

> *[Un]like KDE developers just do it because they want to do it, we [company employees] have clients, the potential users out there. … [T]he application probably needs to be able to work for a lot of different people. And obviously it must be pretty stable.*
>
> (BO0602)

Working Practices in OSS Firms

- **C**entralised Expertise in the Firm, a More Bounded Group

  It's very rare that I [post] question[s] on the Internet. Although I sign [up] for my local Linux community at home, usually I can find the answer to my question on the [firm's] internal mailing list.

  (BO060206)

- **F**ormally Fortified Relationships and Enhanced Trust Between Developers to Solve Problems Efficiently
- **S**uch a Strong Expert-oriented Developing Team is Established to Convince the Customers of the Quality of the Products and Services.

Working Practices in OSS Firms

Mutual Understanding Between Companies and Clients

[We try to tell customers that] it's always better to have open source or free source because it gives [them] a power over the vendor that [they] can never have when it's close source. When it's close source then the vendor can lock [them] in as a user.

(BO0602)

[My client] is a quite clever and open-minded. We understand [each other] perfectly and we know why we do that. He understands that it's better to share. In the case of ERP, he understands that proprietary ERP means you lose complete control on your information system and give to someone else; free software ERP means that you share many things with others that at least you can keep control on your information system. We both think the second case is more perfect.

(JP060202)

The Relationships Between the Community and Firms

I think the company should go closer to the Linux community. We would like to keep some information confidential for business, but we should not forget the open source ideology. Because of open source, so there are we.

(SO060202)

The community is actually necessary. It is. Because that's the form to share interests and to communicate about to get information pass along, in a very broad sense.

(WP060202)

The Relationships Between the Community and Firms

- Bugs-reporting and Patches-contributing
- Social Networking and Social Capitals
- Identity Building

> *[From] the social point of view on free software development, writing creative scripts and sharing them is basic. You get many friends when you do free software. And the relation is not just sharing things on the Internet but making friends with the whole communities of people.*
>
> (JP060204)

Licensing as a Business Strategy

I think [our product] is quite difficult now from the commercial point of view to compete as it is. It makes no sense that customers must pay quite serious money to get source code and they by no means may circulate it. They could just put examples on the web site, saying “here is our stuff, try.” Well, they could put their own stuff but they wouldn’t put that ‘Here is the modified version of the driver’. In fact, if they have put it, I wouldn’t worry too much about it. But the licence said that they could only send to other people when they have licence. So it was constrained. I think it’s quite hard to compete, now, generally speaking, if the source code is not made available. So that’s from the commercial aspect.

(CF060803)

Licensing as a Business Strategy

Another one has to do with you’ve brought as many people as possible if you like. Well, you find that one of the reasons that those become popular even more popular than some other free software such as FreeBSD, is because universities start using it. There’s also this counter culture I suppose to which might attract to universities or university students to use at the first place. But I think it’s important in terms of having as many users as possible. You can do that by intermediary for such as experts done within years licences which said, “You could have this [new product] for research, education, and news.” But you can’t develop politics with it. Perhaps that will do as well. But I think what we got a decision to make certain things possible and I don’t have very good reason to regret it. I think a right thing to do.

(CF060803)

Tensions in the Hybrid Collaboration

"Just For Fun" Shared Interests vs. Money-making Mentality

When you talk about open source, not every software can be open source. It would be Who will pay these people doing something very special? These people who do something very special need to communicate with their users and with people doing competitive stuffs. They need to exchange their ideas. But that doesn't necessary mean this thing has to be open source, why should it?

(WP060202)

[But,] it's an interesting idea, and works well for some projects, doesn't work for other projects. And it doesn't work well for software company which would have a single product.

(WP060203)

Different Views

In the corporate world you would need the mindset that I can make money on this because someone else needs it. That’s been driving a lot of applications, a lot of good software. It still does. I mean that’s the way we distributions compete against each other and write added value to the basic Linux part. It does not work on interested people or motivated people on personal level, but does attract some software developers.

(BO060214)

Freedom of Innovation

[Big firms] don’t want to create new products for new markets. And I know many people they decided to put down a great job as an engineer in a great large corporation to create their own small company doing free software because they are tired of having ideas, which they could never put into practice and share with others. And so that’s one of the reasons also why I am doing my company and the ERP, because if I have an idea, then I can write it as a software then it becomes quickly a product which I can share with others. And even in that new idea can be dangerous for a very old product, I can still do it. So it’s like I have the freedom to innovate, put my innovation on the market, and share it with others. And that freedom does not exist in many large companies. Because there are so much innovation actually in free software, and there are so many people who go that way because that’s probably the only way whenever you have an idea to see in practice used by many people and to improve it after it has been used.

(JP060203)

Different Views

I don’t think that anything like EMACS could have been developed commercially. Businesses have the wrong attitudes. The primary axiom of the commercial world toward users is that they are incompetent, and that if they have any control over their system they will mess it up. [...] The secondary goal is to give managers power over users, because it’s the managers who decide which system to buy, not the users. If a corporate editor has any means for extensibility, they will probably let your manager decide things for you and give you no control at all. For both of these reasons, a company would never have designed an editor with which users could experiment as MIT users did, and they would not have been able to build on the results of the experiments to produce an EMACS.

(Stallman, The EMACS Full-Screen Editor)

Different Views

> *[T]he standard EMACS command language was the result of years of experimentation by many user-maintainers on their own editors … On the fateful day when I gave users the power to redefine their own editor, I didn’t know that it would lead to an earthshaking new editor.*
>
> (Stallman, The EMACS Full-Screen Editor)

The Hybrid Innovation

> **A**s open source software develops, we should expect that it becomes more and more like other dynamic, knowledge intensive industries. In that sense, the dynamics of software development are likely to rely on parallel processes of commercialisation and science. They will rely on both the overall production of public knowledge as well as on the closing off of parts of knowledge production within the firm in order to capture economic value.
>
> (McKelvey 2001: 34)

Heterogeneity and Hybridity

- A Pragmatist Perspective: provides a pluralist domain to contain diverse actors
- Open and tolerant social structure of the FLOSS social world, where ***Us*** and ***Them*** can live together and cooperate, where ***Others*** have the right to express their views, that provide the highest affordance to the technological innovation.
- The FLOSS innovation takes place within such a heterogeneous field, where the local and tacit knowledge can be preserved and borrowed as innovation resources.
- It is also heterogeneity in the social world that affords the hybrid innovation coalescing the innovation resources of the firms and the community.