New Paradigm of R&D Outsourcing

Midori Kato[1], Akitoshi Seike[2], Fujio Niwa[3]

[1] Meisei University, Faculty of Informatics (mkato@mi.meisei-u.ac.jp
[2] Toyama University, Faculty of Economics (seike@eco.toyama-u.ac.jp)
[3] National Graduate Institute for Policy Studies (niwa@grips.ac.jp)

Abstract

The application of the external resource in the corporate R&D has long history in the business world and the research of innovation management.　The accelerating business speed and increasing R&D cost have been varying the applications.　We regard a variety of application of external resource in R&D as "R&D outsourcing." Among them, Open Source projects by firms are stand out prominently in terms of opening the technology absolutely and selecting unspecified and a large number people as R&D partners. This new form of R&D outsourcing is difficult to explain with the existing framework.　Therefore, we introduced the concept that consists of the openness of firms' relationship and the openness of technology to classify R&D outsourcing including the latest cases.

In addition to this, we studied Linux kernel development project as the successful case of Open Source. The reason is that this case covers the prospective three elements in the new paradigm of R&D outsourcing.　Our case study shows the organizational and cultural features of Open Source community from the viewpoint of business administration.　Furthermore, we suggest that the features contribute to the efficiency of R&D project.

Moreover, we discussed the possibility of diffusion of new R&D outsourcing to the field other than software.

1.　Introduction

The application of external resource in the corporate Research and Development (R&D) has been long-studied and discussed.　Now, the necessity which the firms use external R&D resource is increasing more and more because of the accelerating business speed, increasing R&D investment, and accumulating investment risk.　The application of external resource in the corporate R&D has diverse patterns, such as collaboration, Keiretsu, alliance, participation to the consortium.　We use the term "broad or extended R&D outsourcing" to refer all of the applications of external resource in the corporate R&D in this paper.

R&D outsourcing is often discussed from two points of view, the relationship between firms and the influences of that to the innovation.　Mowery=Teece[1] pointed that three forms of cooperative relationship had developed in R&D for last fifteen years, precommercial research consortium, industry-university joint research, and international strategic alliance.　Chatterji[2] classified R&D outsourcing with the axes of "Technology Being Sourced" and "Market being Targeted".

Some companies in Information Technology (IT) industry started "Open Source" projects in 1998, although corporate R&D including R&D outsourcing, should be closed in some extent of the relationship still now.　Open Source is the sophisticated development method of software whose source code and development process are open to everyone, anytime by the voluntary community.　The success of Linux proved its excellence.　Open Source by firms is critically different from conventional R&D outsourcing in the respects that technology is absolutely open and the referral is unspecified number.　In addition, it is an important characteristic of Open Source that they developed most of all on the electronic network.

Robertson=Langlois[3] considered the relation between companies including network and vertical integration and innovation.　NISTEP R&D Out-Sourcing Study Group [4] surveyed the real state of R&D outsourcing of Japanese firms precisely.　Open Source does not go for these researches, and definitely new form of R&D outsourcing.
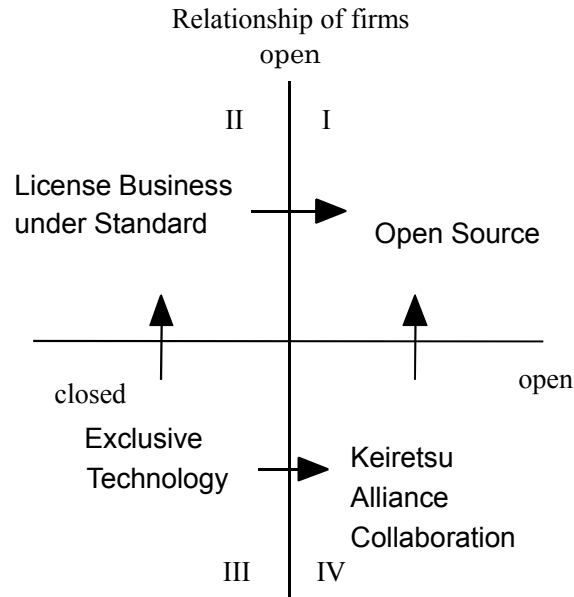
Relationship of firms
open

II | I

License Business
under Standard
→ ▶
Open Source

closed   open

Exclusive
Technology
→ ▶
Keiretsu
Alliance
Collaboration

III | IV

Fig. 1   Classification of R&D with external resource

Therefore, Kato[5] adopted the openness of firms' relationship and the openness of technology as showed Figure 1 to explain R&D outsourcing including conventional types.

In near future, collaboration and finding the "outsourcee" on the network seem to be popular.   Open Source Project by firms has all the three elements, which seem to become properties of R&D outsourcing in the coming new paradigm. That is,

- Open technologies (Open Source R&D)
- Finding partner openly
- Collaboration on the electronic network(exchange/distribution/development of knowledge on the network)

At first, we consider the relationship between the organizational property of Open Source community and its R&D process.   Furthermore, we refer the reason that firms adopt Open Source positively and the possibility of the expansion to the technology fields other than software.

2.   Open Source and R&D strategy of IT companies

2.1 Open Source

Open Source is a general term for developing methods of software whose source code is open to everyone, anytime. The success of Linux kernel development project made Open Source famous to common people.   However, Open Source method and the culture sharing source code have been ordinary among programmers, especially ones for Unix.

2.2 An Experimental analysis of Open Source Community (OSC) 1
--Approach with organizational theory

Here, we take Linux kernel development community as a new R&D model and try to consider experimentally with literatures and open interview, while academic analysis must be based on an in-depth field research.

Linux is Unix compatible Operating System (OS) working on multi platforms.   It is also called PC-Unix because it worked on only Intel chip originally.   Linux, having over 10 million users, has being diffused at far larger growing rate than Unix and Windows NT platform.

Linux development community do most of all activities on its mailing list and open stance to all people.

(1)   Structure of the organization

Raymond [6] likened the conventional development method of software to cathedral, for example, "centralized, a priori approach was required" and "carefully crafted by individual wizards or small bands of mages working in splendid isolation,

with no beta to be released before its time." On one hand, he expressed the development style of Linus Torvalds, who start to development Linux in 1991, as bazaar, "release early and often, delegate everything you can, be open to the point of promiscuity." Allowably, there is hardly limitation of entering and quitting the community. Then again, it does not correspond with flat nor network organization in administrative organization theory. Preferably, we can suggest the hierarchy structure shown as Fig. 2, where the top is Linus Tovals. Under him, core member, contributor, debugger/tester, user/audience (potential volunteer) are in turn. Linus answered to the interview [7] that core members were ten people. Sano[8] showed that the number of contributor, whose source codes were adopted to the kernel, was 273 at version 2.2.5. The layer under contributor debugs and reports the working result of the latest version in their various circumstances as soon as it is released. The lowest layer is Read Only Member (ROM) for the present, and may become contributor or core member.

Roymond [9] told that decision-makers were core member and Linus, and only Linus was responsible for final decision. At least, the structure of authority of decision making is centralized from the viewpoint of business administration, although Raymond described Open Source Community was not centralized.
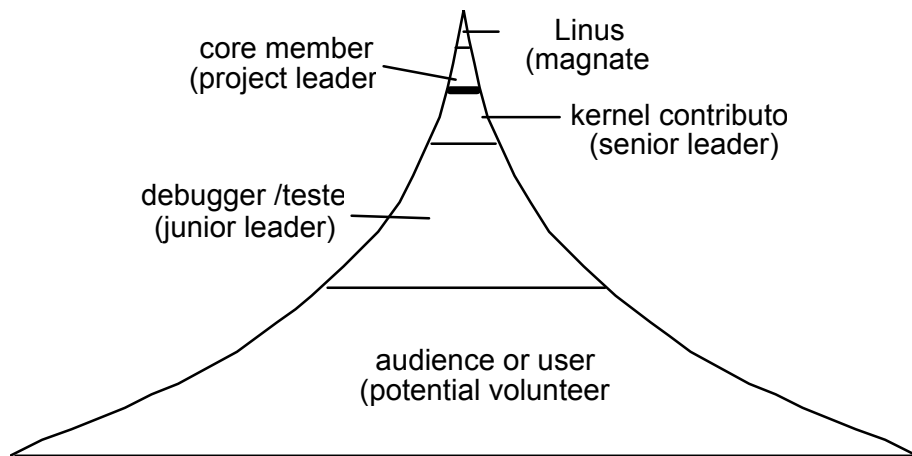

Fig. 2   Hypothetical organization chart of Linux kernel development community

Roymond [9] divided the member of mailing list into core member and periphery. According to him, there is only the communication between core and periphery, and periphery doesn't communicate each other. Linus [7] answered that core members did not have their exclusive mailing list when he had to discuss with them because they rarely had the problem that concerned all core members, having each responsible sub-area (we shall take this point as the efficiency as R&D project later). They describe there is not horizontal communication but vertical one in the community. This picture looks considerably different from the communication that flat or network organizations, which is described to more suitable for knowledge creating than conventional hierarchy's one, presuppose or people assume implicitly on the electronic network.

(2)   Organizational culture

Decision-makers are elected by "peer review" and it is a cardinal rule to praise anyone who does the excellent job. Contributors can have honor at the public space (of course it means mailing list), which is said to motivate them most.   In addition, his/her behavior and the evaluation are open to everyone equally.

The members of the community are sympathizers with Open Source policy and are interested in Linux.   They also adopt the hacker's culture and manner derived from "the internet" and Unix.   Naive people are surely educated by someone sooner or later.   The members are unstinting to give great respect to the programmer whose skills are above some level. The members are highly autonomous and there is conduct code to be autonomous clearly.

To be contributor, he/she does not have to go through complex or inexplicit formality of the organization.   All they have to do is submitting the excellent code to the mailing list.   Cumulative contribution gives them firmer reputation and more accumulation leads them to "promotion" under the consensus of the whole community.

(3)   Management control

In hacker's culture, generally, "`big men' and tribal elders are required to talk softly and humorously deprecate themselves at every turn in order to maintain their status."[9]   Linus's management style is described as " release early and often, delegate everything you can, be open to the point of promiscuity"[6].   He seems to have importance to give not detail instruction but final decision.   Anyway, it is very important for the members that his existence means the powerful vision of

Linux development.

Leader's task may be much less than the one of the ordinary firm's.   Besides, so-called "leader's dilemma" is relieved. One reason is the community's characteristic culture.   For this reason, the evaluation for the leader is very strict.   If the leader does the inappropriate comment or decision over, someone will surely point it, out or superior people will exit. Therefore, the leader will be forced to resign, or the project will be decline.

Management or leadership is necessary for the community, although people tend to describe bazaar development method as the lack of management or leadership.   In the community, the importance of powerful leadership is relatively less because the members are highly autonomous and share the organizational culture.   Especially, management of the motivation is very important.   Linus's management of motivation seems also excellent.   What motivates them is, for example, their satisfaction of the contribution to the community, intellectual interest at high level, honor inside the community.   We think the share of the emotion "pleasure" moves them, especially potential contributor.

Openness is insufficient to explain Linux's success because all of the Open Source communities do not success and Linux is one of the most successful projects in the Open Source communities.   What the organizational structure, culture, communication, and management of the Linux community are very characteristic suggests that the existence of suitable organization and management for Open Source and they contribute the success of R&D project more than a little.   Now, the organization does not come under the conventional ones.

2.3 An Experimental analysis of Open Source Community (OSC) 2   --Efficiency as R&D project

The reason why development speed of Linux kernel is very fast is that the distribution and transfer speed of knowledge or information concerning R&D is considerably fast.   "Open" means that the absolute number of excellent human resource is large if the project is attractive.   The community members understand the technology and the culture deeply and share the development vision.   The knowledge or information is distributed at the tremendous speed openly and its added value is accumulated more and more whenever they bounce off.   Therewith, knowledge whose value is renewed by the minute is shared among (strongly motivated) all the members.

On the one hand, they clarify the hierarchy structure and authority and limit the communication actually in order to avoid the confusion and load of communication, which tend to occur in the team of large number.

The hierarchy of structure is composed reasonably and autonomously to meet the property of each R&D process.   Final decision-maker is one person and each project leader has clear sharing of responsibility.   Decision-makers are very few in comparison with the subscriber of over ten thousand people.   On the other hand, a great number of members concern debug and test process which requires the largest number of people.   The absolute number and variety of test circumstances at this process definitely affect on the quality and development speed of the product.   That is, the community skillfully uses a small number of elite and human-wave tactics appropriately.

Horizontal separation, namely the reasoned modularization is another reason in terms of the organization.   Linus divides kernel into multiple modules and allocates a project leader the responsibility for each module.   Linus's statement [7] suggests his management skill of R&D project.   Appropriate modularization (decentralization) and delegation of power contributes to the prompt development.

That the policy, "release early, release often", works well and results in the prompt bug-fix by the accelerating revolving speed of the mechanism of knowledge generation and distribution without confusion or collision, can be accomplished only by the combination of the organizational structure, the culture, and the management, we suggest.   Probably, rich experience of software development has formulated the rational culture and a code of conduct, and they lead to excellent quality of the product.

3.   New wind of R&D outsourcing

3.1 Open Source R&D
    --Comparison between conventional collaboration and Open Source project by firms

There are two types of Open Source projects by firms project by each firm and cooperative project like Trillian.   Kato [10] showed several cases of both types.   Now, we show the Table 1 as the comparison between conventional collaboration and Open Source project by firms.

Table 1 Comparison between conventional collaboration/ consortium and Open Source project by firms

	Conventional collaboration/consortium	Open Source project by firms
Partner	Specific and a small number(firms)	Unspecific and a large number(community, individual)
Time saving	Good	Excellent
Cost saving	Excellent	Good
Avoidance of exhausting competition	Good	Ineffective
Group making (adding potential users)	Good	Excellent
Image improvement	Ineffective	Excellent
Controlling of the partners	Possible and rather easy	Very difficult sometimes
Object of technological complement	Important and relatively easy to achieve	little
Monopolizing the technology	Possible (monopolization is premise)	Impossible
Standardizing the technology	Possible	Easy to standardize (standardization is premise)
Gain from the technology	Possible, sometimes excellent	Impossible

What is the reason why firms employ Open Source project?   The most important reason that doesn't appear in this table is that the firms establish the business model whose source of profits isn't the technology by the Open Source project. Insistently, from the viewpoint of R&D, the most prospective matter looks like the speed.   Moreover, the expansion of support from the community because of the firm's showing understanding to Open Source, in other words, image improvement and increase of potential user are advantages that conventional collaboration/consortium never have.

3.2 Finding R&D partner on the network

Network can distribute more information at shorter time with less labor than the conventional media can.   Also, physical limitation, for example distance, hardly has influence on the network.   Therefore, it is theoretically possible to adopt worldwide excellent talents if the R&D theme is superior or reward is attractive.   Apparently, finding R&D partner on the network seems advantageous for new or smaller size companies than large firms.   Nevertheless, in our preliminary interview with the existing large firms and small venture companies, we couldn't find the firms who were willing to find the R&D partners on the internet.   Their main reason is that it gains great noise and takes high costs.   They rather prefer conventional method of introduction.

The reason that IT companies selected Open Source community as R&D partners is quite different.   The community originally knows that they should avoid the noise and how to do so in the communication on the network. Furthermore, excellent programmers are authorized by the whole community and they have the system, which omit the people who generate noise.   They have organizational ability that they can select the appropriate method of software R&D.   These IT firms select the community, unspecified and a large number of individual, as R&D partners, however, it helps our better understandings to consider the community as a kind of organization of characteristic culture and ability.

Still, it is precipitant to regard finding R&D partner on the network openly as mere meaningless.   It might be expected that accumulation of firms' information in a field helps firms who find R&D partner.   For example, NIKKEI BOARD GUIDE [11] piles the information of over 100 electronic companies.

3.3 Collaboration on the network

The case study of Linux kernel community shows that in order that collaboration on the network gets better result than conventional one, they have to raise the ratio of knowledge or information, which is possible to be distributed on the network against the knowledge which is necessary for the collaboration.   The higher the ratio is, the more efficient the R&D project must be.

The knowledge or information that can be distributed on the network do not belong to the category of explicit knowledge,

because the community has shared the culture and conduct code, which are known as tacit knowledge. We guess whether the knowledge or information can be distributed on the network or not is definitely important for the collaboration on the network. The conventional framework of knowledge does not seem to explain the new collaboration.

In addition, they have to understand merit and demerit of the communication on the network and learn the communication skill. The knowledge or information of R&D that the network can be distributed and the communication skill are often confused. When we study the collaboration on the network, we should divorce them.

3.4 Expansion of new R&D outsourcing to the industries except for software

The new type of R&D outsourcing is theoretically possible in the fields except software and there is the possibility that it becomes popular in some industries or technology fields. Open Source R&D greatly concerns the firm's business strategy, thus it is not the matter of only R&D. It seems that firms hardly open the emerging standard, for example current DVD-RW, because they will be able to gain much from the standard. In this sense, it may lead the hypothesis of the dynamic relationship of R&D outsourcing types, corporate strategies, and matureness of technology. Meanwhile, when they once open a technology, a group (potential user) may increase explosively. When a firm can hold a dominant position by the prompt standardization or scale economy, Open Source R&D is not a bad option for it. Actually, IBM opened to the public LSI bus architecture, CoreConnect in 1999 [12]. It is strategic aim might be the speedy improvement of the technology and/or enclosing future user (equal to consumer). If firms establish the business model whose source of profits isn't the technology, Open Source R&D is a strategy worth trying.

Matsushita Electric Works opened VOL (Virtual Open Laboratory) at their web site in 1999 [13] and invited idea of new business from the public. Nevertheless, only the result was open and the process was not opened at all. Canon has also the laboratory on the network, while the relationship with R&D partner is closed. Most firms have just started these trials, finding R&D partner or collaboration on the network.

Generally, in the near future, temporary relationship or type of employment will increase, however, it is not perfectly open. Excellent engineers or researchers are likely to attend the other firm's R&D project although they keep belonging to the original firms. R&D under mobilization or openness of human resources must be popular and how to use the network will be the key of successful R&D.

4. Summary and future task

We explored the possibility of R&D outsourcing, which is going into the new paradigm through the case study of Linux kernel development community. Our conclusions of the case study are the following.

- The organization of the community is hierarchical and the structure of decision making is centralized.
- Their communication in the mailing list is customarily limited as part of their system to escape from the noise and confusion.
- The members are autonomous and share their culture and code of conduct.
- Management exists and is effective in the community although the community is pretend far from united.
- The organization, culture, and management contribute greatly the efficiency of the project and the quality of the product.

We suggested three elements for R&D outsourcing in the new paradigm, and we will have the continued studies to generalize them. The expansion of new R&D outsourcing to the industries except for software needs further discussion also from the viewpoint of the knowledge or information that is distributed on the network.

After this, we should have the latest case study and field work to prove our understandings. On the other hand, a large scale of questionnaire to survey the actual status will compensate our study.

References

[1] Mowery and Teece : Engines of Innovation: U.S. Industrial Research at the End of an Era, Harvard Business School Press, 1996.

[2] Chatterji : Accessing External Source of Technology, Research Technology Management, March-April, 1996.

[3] Robertson and Langlois : Innovation, networks, and vertical integration, Research Policy, Vol.24, 1995.

[4] NISTEP R&D Out-Sourcing Study Group (Minkan shikin doukou chousa kentou team) : The new trend of out-

sourcing in Japanese Industrial R&D ("Minkan Kigyo no Kenkyu kaihatsu Out-Sourcing wo chushin to shita atarashii ugoki ni tuite" in Japanese), 1997.

[5] Midori Kato : New strategy of IT industry with open source, 2000 MIS/OA international conference, 2000.

[6] E. S. Raymond : "The Cathedral and the Bazaar", http://www.tuxedo.org/~esr/writings/cathedral-bazaar/.

[7] Nikkei Computer : Shogyoteki de nai kaihatus sutairu koso honraino arubeki sugatada, Linux perfect guide for corporate users,1999.

[8] Sano Taketoshi : Comment kara yomu Linux kernel，http://www.linux.or.jp/JF/JFdocs/readkernel.html.

[9] E. S. Raymond : "Homesteading the Noosphere", http://www.tuxedo.org/~esr/writings/homesteading/.

[10] Midori Kato : A study of Open Source Community with organizational theory and the affect that the community gives to the corporate R&D strategy, NISTEP discussion paper, in the press.

[11] Nikkei Business Publications, Inc : http://ne.nikkeibp.co.jp/board/inq/.

[12] IBM : CoreConnect Bus Architecture, http://www.chips.ibm.com/products/powerpc/cores/crcon_pb.html.

[13] Matsushita Electric Works : NAIS Open, http://www.mew.co.jp/opn/opn.html.