

A Comparison Study on the Coordination Patterns in Oss Community

Shinsuke Matsumoto, Yasutaka Kamei, Masao Ohira and Ken-ichi Matsumoto
Graduate School of Information Science, Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara, Japan
{shinsuke-m, yasuta-k, masao, matumoto}@is.naist.jp

ABSTRACT

[illegible]

Categories and Subject Descriptors

H.5.3 [Information Interfaces and Presentation]: Group and Organization Interfaces—Collaborative computing, Computer-supported cooperative work, Web-based interaction

General Terms

HUMAN FACTORS

Keywords

FOSS, Social Network Analysis

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
Copyright 200X ACM X-XXXXX-XX-X/XX/XX ...\$5.00.

1. INTRODUCTION

高品質かつ高機能なソフトウェアの開発を続ける OSS コミュニティが注目を浴びるようになっている。OSS は世界各地の開発者がネットワーク上でコミュニケーションをとりながら共同開発を行うという、分散型開発プロジェクトの一例である。OSS 開発のコミュニケーション構造の実態を分析することで、OSS 開発のみならず現在盛んに行われつつあるオフショア開発などの分散開発環境に対する有益な知見が得られると期待できる。

開発者同士のコミュニケーションの実態を分析した研究が多数報告されている。HowisonらはOSSコミュニティ内の開発者同士が形成する informal social structure について分析を行った [3]。According to them, 多くの OSS コミュニティでは開発者コミュニティの中で中心的な役割を担う開発者が、長期に渡って存在していることが明らかとなった。Bird et al. examined the correlation between centralities and Apache developer contributions from developer mailing lists and the changed history of source code[1]. The analysis results indicated that developers communicating with many other developers contributed more to source code changes.

また、OSS 開発者のコミュニケーションを対象とした研究が行われる一方で、OSS 開発の重要な役割の一つとしてユーザの存在についても指摘されている。Raymond らは開発されたソフトウェアを利用するユーザを OSS コミュニティにおける共同開発者と位置づけ、その重要性を指摘した [4]。多くのユーザがソフトウェアを利用することにより、コードに含まれるバグが指摘され、結果として品質の改良にも繋がるためである。また、Ye らは OSS コミュニティのユーザがコミュニティ参加者同士での学習を通じてコミュニティへの帰属意識を持ち、次第に重要な役割へと変化するという参加形態（正統的周辺参加）が開発者のモチベーションに繋がるとしている [5]。

このように開発者とユーザの協調作業の重要性が指摘されている一方で，“How structure of an informal social structure formed by users and developers in an OSS community.”という点については明らかにされていない。そこで、本稿ではOSSコミュニティにおける開発者とユーザの構築する informal social structure について、2つのコミュニティ（開発者、ユーザ）に所属し、仲介役を担っていると考えられる人物に着目し分析を行う。具体的には、「informal social structure の中心的な人物はだれか?」「開発者、ユーザの両方のサブコミュニティに属する人物はどのような役割を担うか?」という点について着目する。

分析では一般に良く知られた 2 つの OSS コミュニティを対象として、ある一期間における informal social structure の詳細についてと、時間の経過に伴う構造の変化という 2 つの観点から分析を行う。

This paper is organized as follows, ...

2. RELATED WORK

[コピー含]

OSS コミュニティの開発形態を分析した事例が、数多く報告されている[?, ?, ?]. Mockus らは、これまで OSS コミュニティでの開発について指摘されていた事柄の正しさについて、CVS データ、バグ報告データから調査し、その結果を報告した。分析では、開発者全体の人数、主な開発者の人数、バグの含有密度、バグの修正期間などに着目している。Jensen らは、OSS 開発者の role migration や project career advancement process について、3 つの OSS 開発プロジェクトを比較し、その違いについて報告を行っている。これらに対して我々は、開発者とユーザの形成する informal social structure について分析を行い、その調査結果を報告する。

また我々の分析と同様に、OSS 参加者の形成する informal social structure を一つのネットワーク構造と見なし、その構造について分析した研究も行われている。Hossain らは、Enron 社内で用いられていた email の履歴データから、テキストマイニングと中心性に基づくネットワーク分析を行い、社員 (employee) 間のコーディネーション能力について分析した。結果、コーディネーション能力の高い社員は、高い中心性を持つことが明らかとなった。Howison らは、バグ報告スレッドの送受信関係をネットワーク構造として、バグ報告者と中心性との関係について分析を行った。5 つのプロジェクトを扱ったケーススタディにより、巨大なプロジェクトほど中心的な役割を担う開発者が長期に渡って存在していることが明らかとなった。本稿ではこれらの分析方法と同様に、オンラインコミュニケーション履歴データから参加者同士のコミュニケーションネットワーク構造を抽出し分析の対象とする。

これまでに我々と同様、OSS コミュニティを中心性の観点から分析した研究はいくつか報告されている[?][3][1]. Yamauchi らは、開発者の中心性と調整行動との関係をメーリングリストデータから調査した結果を報告している[?]. ケーススタディでは、FreeBSD Newconfig と GNU GCC プロジェクト、2 つの事例を紹介し、開発者間の調整行動を行っている人物が存在することを明らかにした。また、Howison らは、バグ報告者の人数と中心性との関係を、バグ報告データから調査した結果を報告している[3]. ケーススタディでは、SF.net に登録されている OSS コミュニティ、5 つの事例を紹介し、コミュニティの大半で、相当長い期間コミュニティの中心である特定の人物が存在することを明らかにした。特に、Bird らは、我々と同様 Apache コミュニティを対象として、ソースコードの更新履歴から求めた開発者の活動の度合と開発者の中心性との相関を調査した結果を報告している[1]. ケーススタディでは、多くの開発者と交流している開発者ほど、活動の程度が高いことを明らかにした。

- dev mail Ap vs Moz: Moccas Hervsleb ICSE 2007: 開発者昇格プロセス jensen SCACCI

- network howison bird

- ユーザがどのように

3. INFORMAL SOCIAL STRUCTURE IN OSS COMMUNITY

Since, in general, participants in FOSS communities are geographically-distributed, they rely heavily on online media as a way to communicate each other. So the history of communications among participants on online media such as mailing lists news groups and bulletin board systems helps us extract the informal social structure of participants in FOSS communities. In this paper, we define the informal social structure by sender-receiver relationships on online media. Figure 1 is an example of the informal social structure in FOSS community. The left side in Figure 1 shows

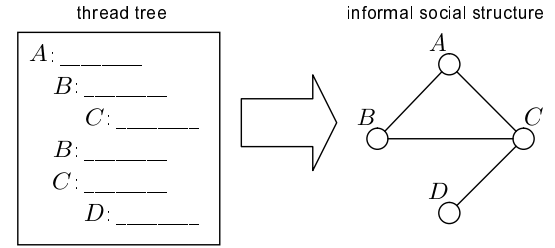


Figure 1: Constructing an informal social structure.

a thread tree in online media and the right side is a network structure constructed from the thread tree, regarding nodes and edges as message senders and replies respectively. For example, if a participant B and C replies to a message sent by participant A , edges are linked from node B and C to node A . In this way, we can represent and analyze the informal social structure in FOSS communities as network structures. #ひとまず冠詞のチェックは後回し。

In FOSS community, there are some roles of participants (e.g. developer, tester and users), and subgroups according to the roles of participants. Through using online media, technical matters (for instance, topics related to new features and/or defects of FOSS products) are discussed in a developer group, Q&As about usage of software are discussed in user group, and so forth. Figure 2 illustrates an example of the informal social structure formed by two subgroups in FOSS community. P_{dev} (yellow node) is a participant who had sent a message to the developer group (the left circle in the figure), P_{usr} (blue) is a participant who had sent a message to the user group (the right circle in the figure), and $P_{d \cap u}$ is a participant who had sent to both the developer group and user group. By discussing topics corresponding to the role of each subgroup, participants of subgroups can efficiently obtain relevant information for themselves. However, for instance, even if P_{usr} in the user group discussed useful ideas which might improve the quality of software products, P_{dev} in the developer group could not notice the ideas. As the result, the ideas would be buried in communication logs of the user group. Therefore, $P_{d \cap u}$ who participates in the both groups has an important role of transferring feedbacks from users such as feature requests and/or defect reports to developers. In parallel, $P_{d \cap u}$ also have a role of user support because have s/he are knowledgeable about developed software as a developer. We consider that $P_{d \cap u}$ who transfers information discussed in one group to other group largely involves in facilitating communication and collaboration among participants in FOSS community, that is, $P_{d \cap u}$ have strong influence on the informal social structure. In this paper, we focus on the role and network structure of $P_{d \cap u}$ based on the reasons described above.

4. ANALYSIS METHODOLOGY

4.1 Perspective of Analysis

We can analyze the informal social structure formed by FOSS, based on the following two perspectives on complex social networks.

4.1.1 Static Perspective

By statically capturing the informal social structure at

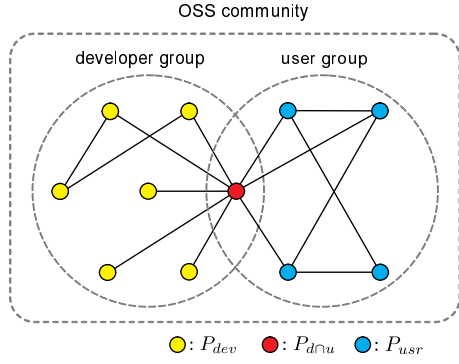


Figure 2: An example of FOSS community.

some point in time, we can examine the state and characteristics of the structure in detail. In this perspective, the informal social structure can be analyzed by using visual representations of the structure and previously proposed metrics. In this paper, we analyze characteristics of the informal social structure itself and the coordination ability of individuals in the structure.

Furthermore, FOSS community such as Figure 2 can be seen to have three kinds of informal social structure. Analyzing $P_{d \cap u}$ in each network, we can better understand the role of $P_{d \cap u}$ which bridges between developers and users.

- all participants network formed by $P_{d \cap u}$, P_{dev} and P_{usr} (OSS community in Figure 2)
- developers network formed by $P_{d \cap u}$ and P_{dev} (developer group in Figure 2)
- users network formed by $P_{d \cap u}$ and P_{usr} (user group in Figure 2)

For static analysis of the informal social structure in FOSS community, we use the average path length L and the clustering coefficient C which are widely used for measuring the feature of complex network structure (L and C are described in Section 4.2).

We can also use Freeman's centralities[?] as a metric to see the ability of coordination. As Hossain et al. [2] reported the analysis results of the social structure in a large company based on email data, there is a significant correlation between coordination ability and Freeman's centralities (Degree, Betweenness, and Closeness). Hossain et al. also found the highest correlation between coordination ability and Betweenness Centrality, which is one of Freeman's centralities and a metric for indicating the degree of intermediation between people.

However, Howison et al. [?] argued that Betweenness was not suitable for analyzing the informal social structure constructed using mailing lists data or news group data, because such the data are open to public so that people can share information (i.e., a person does not need to transfer information from one person to another person). For this reason, Howison et al. used Degree Centrality for their analysis of FOSS communities. Degree Centrality is a value of normalized degrees (the number of edges) by network size (the number of nodes). Degree Centrality is essentially same as degrees when comparing two nodes in a same network or

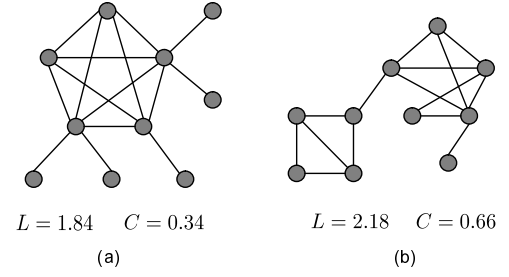


Figure 3: An example of analysis metrics.

same scale of network. For simplicity, in this paper, we use degrees as a metric for the coordination ability.

4.1.2 Dynamic Perspective

Anyone can freely participate in FOSS community and they can freely discuss and exchange ideas on equal footing. For this reason, the informal social structure formed by participant in FOSS community constantly changes over time. In order to analyze this living network, not only the static analysis described earlier but also the dynamic, time series analysis is required. In the dynamic perspective on networks, we analyze changes of the informal social structure and the coordination ability of $P_{d \cap u}$ over time. Since it was difficult to analyze all $P_{d \cap u}$ in large-scale networks in all period of time, we selected top 5 $P_{d \cap u}$ with the highest degrees and analyzed them in time series.

4.2 Network Analysis Metrics

This section describes the average path length L and the clustering coefficient C which are used in our analysis.

4.2.1 Average Path Length

The average path length L is defined as the average number of steps along the shortest paths for all reachable pairs of nodes. When the average path length of node v_i is L_i and the number of nodes is n , L is defined as follows.

$$L = \frac{1}{n} \sum_{i=1}^n L_i. \quad (1)$$

Generally, a network with small L is highly efficient in distributing information. However, since the informal social structure targeted in this paper is defined using data from open media such as mailing list, it differs from general information transfer networks. Figure 3 is an example of L and C of two networks. while there seem to be intensive discussions among five participants in Figure 3(a), although in Figure 3(b) there seem to be diverse discussions and the value of L is low. In this paper, we consider L shows not the efficiency of information transfer but the degree of concentration of discussions among participants.

4.2.2 Clustering Coefficient

The clustering coefficient C_i of node v_i is the proportion of links between two nodes which are linked to v_i , this means C_i is the proportion of triangles among any three nodes. The clustering coefficient C of the whole network is the average number of C_i defined as follows.

$$C = \frac{1}{n} \sum_{i=1}^n C_i. \quad (2)$$

In Figure 3(a), there are five participants that has only one edge (they do not have triangles with other participants). Because C_i of participants who linked with them are low, C of the network (a) is lower than (B) as the results. Although the network is separated into two clusters in case of Figure 3(b), the ratio of nodes connected each other is high and then C of the network (B) is higher.

In this paper, we consider C shows how many participants know each other. Note that, although we calculate C to the developer network and the user network, we does not calculate C to the all participants network. C is not suitable for applying to a network that contains $P_{d \cap u}$, P_{dev} and P_{usr} as shown in Figure 2, because P_{dev} and P_{usr} belong to a different network, that is, an direct edge between P_{dev} and P_{usr} (a triangle of $P_{d \cap u}$ and P_{dev} and P_{usr}) does not exist in a practical sense.

5. CASE STUDY

In this section, we describe the case study which used communication history data of two well-known FOSS community.

5.1 分析対象 OSS コミュニティ

In this paper, we compare two informal social structures of two FOSS communities that are Apache HTTP Server and Netscape Browser. Two softwares developed by the FOSS communities are described as follows.

- Apache HTTP Server¹:

In February 2008, Apache has a dominant share of 50.6% in the world's HTTP server market[?]. Apache community develops three major version in parallel, and they continue to release many security patches. The community is widely recognized as high quality and high reliability software

- Netscape²:

Netscape is a web browser software which dominated 80 % of browser market at peak in 1996. Although following the research of Raymond[], Netscape community decided to open its source code in order to oppose expansion of Internet Explorer in 1998, the community could not claw back market share. Software development and user support of Netscape were ended on March 1st, 2008.

5.2 Data Source

History of communications were collected by online media that was used in public discussion of each FOSS community. Apache used mailing list and Netscape used newsgroup. We consider that there is little difference between these medias for analysis of informal social structure as follows. The difference of mailing list and newsgroup is how to receive messages. In the case of mailing list, messages are automatically distributed to mailbox if user registered its mail address beforehand. In the case of newsgroup, users need to choose

¹<http://httpd.apache.org/>

²<http://browser.netscape.com/>

Table 1: Quantity of each perspective.

		apache	netscape
perspective 1	time	2005/12	2002/08
	# participants	683	1,058
	# mails	4,068	3,509
perspective 2	begin	2001/11	1999/09
	end	2007/09	2007/02
	# participants	12,710	23,386
	# mails	112,143	91,363

Table 2: result all metrics.

		Apache ($P_{d \cap u}$)	Netscape ($P_{d \cap u}$)
all	# nodes	661 (24)	1,022 (38)
	# edges	1230 (720)	1,600 (221)
	$\langle k \rangle$	1.86	1.57
	L	3.34	4.58
	C	—	—
dev	# nodes	131 (24)	319 (38)
	# edges	431 (302)	375 (121)
	$\langle k \rangle$	3.29	1.18
	L	2.76	6.64
	C	0.21	0.04
user	# nodes	554 (24)	741 (38)
	# edges	861 (540)	1246 (154)
	$\langle k \rangle$	1.55	1.68
	L	3.31	3.11
	C	0.08	0.10

and receive a message actively. So, there is little difference between discussion media because informal social structure is formed by action of message sending. Moreover, (analysis period of time 分析期間??) coincides with decadent of newsgroup, however there is no publications that Netscape community shifted to other media.

Table 1 shows number of unique participants and number of mails in each period of time of perspective 1 and 2. The period of time of perspective 1 is time when the major version was released (Apache: ver.2.2.0, Netscape: ver.7.0). We use all periods that could collect history data opened on Web in perspective 2. Since there is many discussions that are ongoing over months, we consider a informal social structure of month m is formed by history data of month m and $m \pm 1$.

5.3 Result

5.3.1 Result of the Perspective1: static

We describe the result of perspective 1 that performed static analysis. Pajek³ that is a tool for large network analysis was used for visualizing networks and calculation of metrics. As a graphing algorithm of a network, we used Kamada-Kawai algorithm which is spring layout algorithm. Note that, each drawn network includes no nodes that have no edge (he/she sent a message but there was no reply) for improving the visibility because each network is big-scale.

Informal social structures of each FOSS community are shown in Figure 4. The left side in Figure 4 shows Apache and the right side shows Netscape. The top shows all participants network ($P_{d \cap u}$, P_{dev} , and P_{usr}), the center shows

³<http://vlado.fmf.uni-lj.si/pub/networks/pajek/>

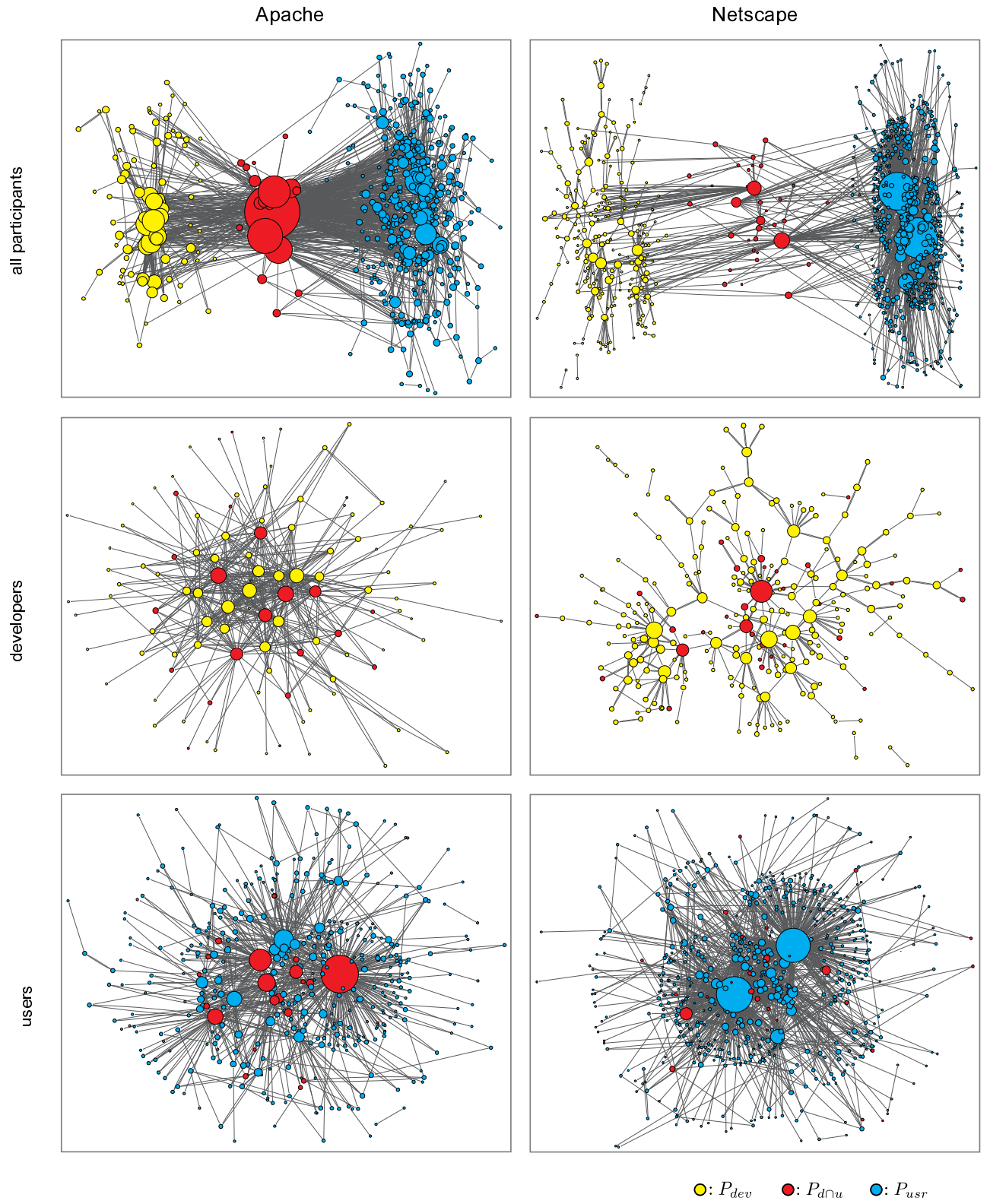


Figure 4: result all network.

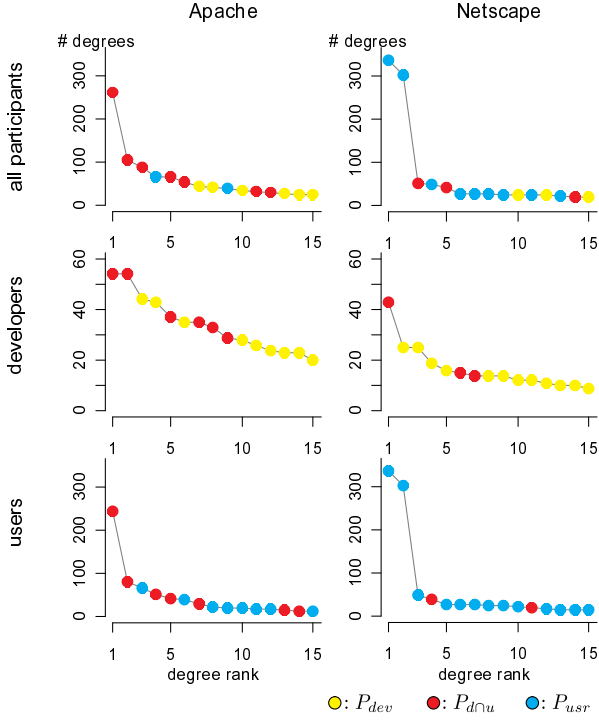


Figure 5: result all degree.

developers network (P_{dru} and P_{dev}), the down side shows users network (P_{dru} and P_{usr}). The size of nodes means number of degrees, the color means participant role same as Figure 2. The metrics which shows feature of each network structure are shown in Table 2. The values in parenthesis show only about P_{dru} . Degrees of each node in coordination ability (number of degrees) order are shown in Figure 5 for confirming the specific number of degree, although participants who has high coordination ability (has many degrees) of each network are visible in Figure 4. The x-axis shows the number of degrees, the y-axis shows ranking of degree. The layout of graphs and the colors of points are same as Figure 4.

First, we describe the difference of all participants network between Apache and Netscape (in the top side of Figure 4). The size of network (the number of nodes and edges) of Netscape is larger than Apache in Table 2. The number of P_{dru} of Apache is less than Netscape (Apache: 24, Netscape: 34). However, there is more edges of P_{dru} of Apache (Apache: 720, Netscape: 221) and many messages were sent from per P_{dru} in the case of Apache community. The result is visible in Figure 4.

Next, we describe the difference of developer (P_{dru} and P_{dev}) network between Apache and Netscape (in the top side of Figure 4). In Table 2, although the number of developers of Apache is half of Netscape, Apache has many edges and average edges $\langle k \rangle$ is twice as Netscape. The average path length L of Apache is the smallest than all other networks, clustering coefficient C is the largest referable to the large $\langle k \rangle$. On the other hand in Netscape, L is the largest and C is the smallest than all other networks. In apache developer network, many big nodes which has many edges are high density and small nodes which has few edges are connected in the figure outside. The network looks like star network in

broad perspective. Meanwhile, Netscape developer network structure formed by many little stars which includes about little 10 nodes and one big node concatenated. In other words, developer networks have different structure between Apache and Netscape. The number of P_{dru} in high degree of top 15 nodes is Apache is six and Netscape is three in Figure 5.

Finally, we focus on users (P_{dru} and P_{usr}) network. The difference of the structure of user network cannot be confirmed in Table 2, because there is little difference of $\langle k \rangle$, L and C . However, in Figure 4, Apache user network confirms a structure that has the highest degree P_{dru} centered cluster and some high degree P_{dru} are situated at the cluster. In contrast, Netscape user network confirmed by big P_{usr} centered two large clusters. These results are shown in Figure 5 too. Moreover, the number of P_{dru} are only two in high degree of top 15 nodes in Netscape user network. In other words, user networks differ in not structure but role of center participant.

5.3.2 Result of the Perspective2

The result of perspective 2 that analysis transition of informal social structure. Changing of metrics of all participants network are shown in Figure 6. The left side shows Apache and right side shows Netscape, the upper side is the number of edges and nodes, the down side is the average number of edges $\langle k \rangle$ and L . The vertical dashed line means in the term of perspective 1.

The number of edges, L and C changed little in the all terms, although the number of nodes of Apache network reached a peak in about 2002 and gradually decrease after that. In other words, Apache network structure is equable in size and structure. In perspective 1, we are considered to have analyzed the term whose network was equable.

In contrast, the number of nodes and edges of Netscape network are decreasing in the all terms. Especially, since $\langle k \rangle$ is gradually increasing from 2000 to 2004, nodes with many degrees remained instead of nodes with few edges decreasing. In perspective 1, we are considered to have analyzed the term whose network was decreasing.

Next we describe the changing of rank of top five P_{dru} of many degree. PN_{dru} means top N of high degree P_{dru} in perspective 1. Figure 7 shows change of rank of PN_{dru} ($N = \{1, 2, \dots, 5\}$).

In Figure 7, $P1_{dru}$ is always on top in all periods for about six years in Apache community. The top two and three are occupied by $P2_{dru}$, $P3_{dru}$, $P4_{dru}$ and $P5_{dru}$ after 2004. As a result, forming the equable network structure was confirmed in perspective 1, P_{dru} who links two networks with high communication ability are also little changed in all terms by perspective 2.

In contrast, $P1_{dru}$ of Netscape community keeps high degree rank only around from 2002 to 2003. $P2_{dru}$ frequently replaces terms to many speak or terms to little speak because rank of $P2_{dru}$ fluctuates widely. $P3_{dru}$, $P4_{dru}$ and $P5_{dru}$ are shown in the graph only five months. These result means there is no P_{dru} who has high communication ability over time with decreasing the scale of Netscape community.

6. DISCUSSION

6.1 *****

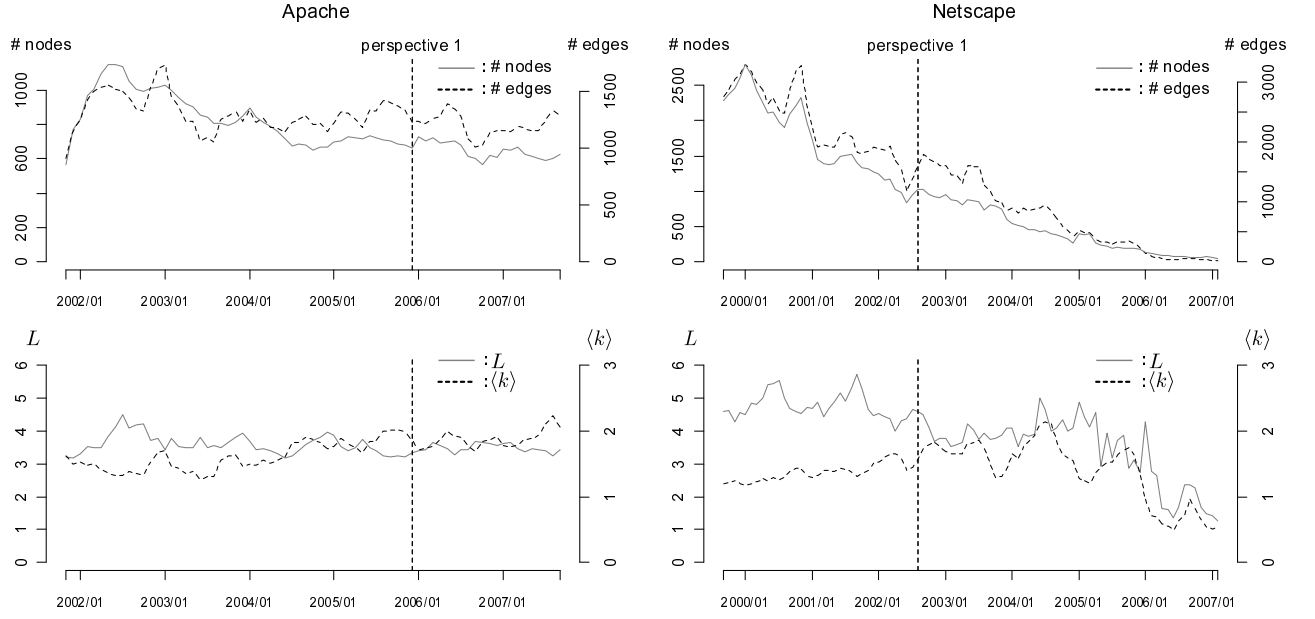


Figure 6: result graph metrics.

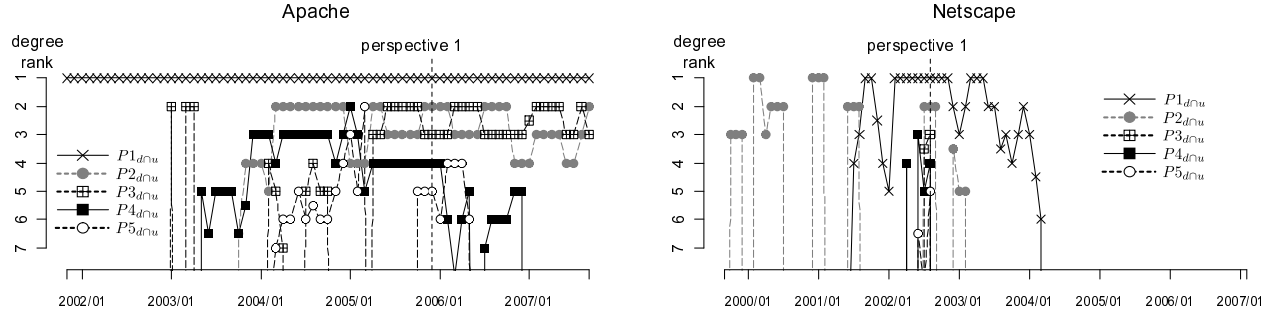


Figure 7: result degree ranking.

The differences between Apache and Netscape by the analyses of perspective 1 and 2 are 1) developer networks differ in network structure, 2) user networks differ in type of center participant, 3) all participants networks differ in P_{dru} who has high communication ability with long term. This section discusses the three result.

Apache developer network has high C and especially density of about 30 developers is very high. It can be seen that high motivation developers discuss aggressively each other. In Netscape developer network structure is more Q&A structure (this means structure of high degree nodes connected by low degree node) than discussion structure. The result would lead to discussion does not concentrate, two communication collaboration in Netscape developer communication structure is difficult because there is no P_{dru} who has high communication ability.

In the Apache user network, the ratio of P_{dru} in high degree participants who answers users question actively are high, and the network forms a structure which P_{dru} absorbs or transfers the users comments or ideas to developer community smoothly. The user network of Netscape has little P_{dru} and formed by P_{usr} centered structure. These result indicates Apache network has an advantage for

The case of all participants network, P_{dru} in Apache network communicate with P_{dev} and P_{usr} aggressively than P_{dru} of Netscape, they create a better relationship with developers and users. Additionally there is $P1_{dru}$ who has always the highest degree, he/she seems as powerful reader who preserves high motivation. The result confirms to Howison's research[?] that analyze developer community.

We conclude a high and long-term motivated participant as Linus and participant who collaborates closely with two communities are key factors to success of geographically distributed software development as a FOSS community.

6.2 Threats of Validity

In this paper, we considered that message transfers/receivers whose mail address corresponds completely were a same person in communication history data. However, there is some persons who use some mail addresses or changes one's mail address. So, we will need to analyze using a mail processing method that uses before @ characters and consider first name or last name[].

We considered that an informal social structure is an undirected graph in the analysis, however communication includes directed relationship as messages sender/receiver. We are going to perform analysis which considers a direct edges

like research of Bird et al]].

Note that, we compared that FOSS communities that develops different types of software. Though Apache community develops web server software and the users have a certain level of knowledge, Netscape is Web browser software and the users are from beginners to experts. The difference in types of users could affect the analysis of $P_{d \cap u}$.

7. CONCLUSION AND FUTURE WORK

8. ACKNOWLEDGEMENTS

This research is being conducted as a part of the EASE project in Comprehensive Development of e-Society Foundation Software program, and Stage Project, the Development of Next Generation IT Infrastructure, and Grant-in-aid for Scientific Research (B) 17300007, 2007 and for Young Scientists (B), 17700111, 2007, by the Ministry of Education, Culture, Sports, Science and Technology of Japan.

9. REFERENCES

- [1] C. Bird, A. Gourley, P. Devanbu, M. Gertz, and A. Swaminathan. Mining email social networks. In In Proceedings of the 2006 International Workshop on Mining Software Repositories (MSR'06), pages 137–143, 2006.
- [2] L. Hossain, A. Wu, and K. K. S. Chung. Actor centrality correlates to project based coordination. In In Proceedings of the 2006 Conference on Computer Supported Cooperative Work (CSCW'06), pages 363–372, 2006.
- [3] J. Howison, K. Inoue, and K. Crowston. Social dynamics of free and open source team communications. In In Proceedings of the 2nd International Conference on Open Source Systems (OSS'06), pages 319–330, 2006.
- [4] E. S. Raymond. The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary. O'Reilly and Associates, 1999.
- [5] Y. Ye and K. Kishida. Toward an understanding of the motivation open source software developers. In In Proceedings of the 25th International Conference on Software Engineering (ICSE'03), pages 419–429, 2003.