

オープンソースソフトウェア開発におけるイノベーションの分析

山内 裕

横澤 誠

篠原 健

京都大学情報学研究科社会情報学専攻

世界中のプログラマが広域ネットワーク上で協調しながら開発しているオープンソースソフトウェアは、企業で開発されるソフトウェアと競争するまでに至っている。しかしながら、このオープンソース開発が強く依存している電子メディアは、従来の開発のような対面での接触に比べて多くの欠点を抱えている。それではどのようにしてこれらの欠点を克服し、さらに競争優位へと結び付けているのであろうか？本研究は、あるプロジェクトに注目し、その開発を観察することで三つの事実を発見した。(1) 協調より行動を指向している。(2) 仕様の議論は実装とオーバラップしている。(3) 確立された技術の結合には電子メディアが適している。

Analysis of Innovations in Open Source Software Development

Yutaka Yamauchi

Makoto Yokozawa

Takeshi Shinohara

Department of Social Informatics,
Graduate School of Informatics, Kyoto University

Open-source software, which is developed over the Internet by physically dispersed programmers, competes with that developed in firms. The electronic media on which open-source development heavily relies, however, are more limited than face-to-face contacts in firms. Then, how can they overcome the limitations and build their competitive advantages? In this paper, we present three findings based on the observation of a open-source project: (1) bias for action rather than coordination, (2) overlap of specification settlement and implementation, (3) effective combination of technologies.

1 はじめに

今日、世界中に散らばったプログラマがインターネットを介して協調し、企業と遜色のない製品を作り出している。このようなオープンソースソフトウェアは、協調作業やイノベーションに適さない限定的な電子メディアに強く依存していることを考えると、驚くべき成果である。本研究はオープンソース開発のメディア利用に注目することで、その競争優位の源泉を探究した。

オープンソース開発では、ほとんどの組織化活動を電子メール上のコミュニケーションによって行っており、イノベーションの大部分は電子メールのアーカイブを観察、分析することで明らかにすることができた。メールを分析すると同時に、メールには現れてこない組織化活動を捉えることや、分析の妥当性について意見を得るために、参加者のインタビューを行った。

2 オープンソース開発と電子メディア

2.1 オープンソース開発組織

通常、オープンソースソフトウェアを説明するときに、次の3つの特徴を考える。(1) 開発者は地理的に分散した不特定多数であり、(2) インターネットに代表される広域情報ネットワークを介して協調し、(3) ソースコードがオープンである。ソースコードがオープンであるとは、ソフトウェアのソースコードを、誰でも変更し配布することができるように、著作権上の制限を緩めた、あるいは無くしたということを指す。Linux, Emacs, Perl などはその例である。

最近のオープンソース開発の成功は、その定義通り「オープンなソースコード」に帰するところが多い。例えばLinuxでは、ソースコードをオープンにすることによって、例え不完全でも早めにソースコードを公開し、問題が生じればユーザに改善してもらおうという形態の協調作業が可能となる。そのため、周辺機器

メーカーからのドライバのサポートがほとんどないにも関わらず、新しい機器をサポートするのが早く、また、新しい技術や機能の導入やバグの除去が早い。さらに、ソースコードに変更を加え独自の設定を行えるために、組み込みシステムや実験環境などにも導入されている。

2.2 電子メディアの欠点

対面に比べて、電子メディアでは社会的存在感 (social presence) の伝達が弱くなる [9]。そのため、たとえ伝達する情報量が同じでも、内容についての確信は異なる。テキストによるコミュニケーションよりも、音声、映像のあるコミュニケーションの方が社会的存在感は大きくなる。

電子メディアは対面接触と比べて一方的に劣っているのではなく、トレードオフの関係にある。電子メディアは情報の量の伝達には対面よりも適しているのに対して、情報の意味の伝達には向かない [3]。例えば、「試験の評価に抗議する」には対面が適しており、「試験の日時や場所の案内」には電子メディアが適している。ここで情報の意味のあいまいさは多義性、情報量の不足は不確実性と呼ばれる。

電子メディアは多義性解決能力 (リッチネス; richness) によりメディア選択に影響を与える [8]。優秀な管理者ほど、多義的なメッセージはリッチなメディアを、多義的でないメッセージにはリッチでない (リーン; lean) メディアを利用する。また、同様に社会的な効果として、電子メディアは肩書き、性別、年齢などの社会的文脈 (social context) が抜け落ちるために、人々は自己中心的になり、日常押さえられていた感情的な行為が頻出する (flaming) ようになる [10]。

以上の議論から、電子メディアは速いインタラクションを必要とするような複雑な協調作業には不向きであり、イノベーションに必要な直感や雰囲気といった暗黙的情報の伝達には適さないということがわかる。オープンソース開発が、このような限定的なメディアを利用しているにも関わらず、スムーズな協調が可能であり、イノベーションを実践できるのはなぜか？ 以下、この問題に答えていく。

3 研究の対象: Newconfig プロジェクト

調査対象のオープンソース開発組織として FreeBSD コミュニティの Newconfig プロジェクトを採用した。この組織は、一定の目的のために最初からある程度の人を集めて始まったものであり、Linux のように一人が開発し他の人がそれに拡張を加えるというものではないという点で、従来の企業におけるプロジェク

トと共通する部分が多い。1998 年 6 月から始まり 1999 年 6 月に一応の締め括りを迎えた。参加者は全員日本人であるが、地理的には分散している。

FreeBSD とは、UNIX と呼ばれる基本ソフトウェア (オペレーティングシステム) であり、主に PC 上で動作する。ソースコードはオープンであり、誰でも開発に参加できる。Newconfig プロジェクトの目的は、従来から構造が問題となっていたデバイスを管理する部分 (config) を、新しい構造のもの (config.new) に更新しようというものであり、FreeBSD のソースコードを大幅に変更しなければならず、その作業は大規模なものとなる。

4 Newconfig プロジェクトにおけるコミュニケーションパターン

コミュニケーションを物理的なデータとしてではなく、シンボリックな意味の集まりであると捉えることが重要である。そのために本研究では一貫して、「内容分析」の手法 [7] を用いる。内容分析とは、データをもとにそこから (それが組み込まれた) 文脈に関して再現可能でかつ妥当な推論を行うための一つの調査技法である [7]。分析単位としては、電子メールメッセージ全体ではなく、それに含まれる一つ一つの「段落」を考えた。これは、一つの電子メールメッセージ中に、複数の独立した議題が含まれることが多く、これらを別の単位として扱いたいからである。これら個々の単位を「メッセージ類型」に分類する。メッセージ類型は、「質問」、「提案」などのそのメッセージの種類を表わす名義尺度である。メッセージ類型の詳細と集計を表 1 に示す。再現性の指標である Cohen のカッパ係数は、「意見」、「説明」、「依頼」では 0.8 を下回っているが、できるだけ細かく類型を分類しようとしたためにコードが多少混乱したためである。以下の分析では、これらの信頼性の低い類型は、あまり重要な位置を占めないものでそれほど問題ではない。

この時点で目につくのは、「追加」というメッセージ類型の存在である。当初、この類型を想定していなかったときは、これに分類されたメッセージを何に分類すべきかははっきりしなかった。単に、「情報が抜け落ちているのでそれを提示する」というだけの目的であり、「回答」、「提案」にも分類できなかった。このようなメッセージがあまりに多いために、新しく「追加」というメッセージ類型を設定した。

次に、議論は議題ごとにまとまりがあり、一つの議論が終了すると次の議題の議論が始まる。このようなひとまとめの議論をスレッドと呼び、どのような

表 1: メッセージ類型

メッセージ類型	詳細	出現数, 頻度 %	一致率 (Cohen の カッパ)
回答	質問に対して答えを持っているときそれを伝える。	103, 18.2	.906
質問	はっきりした答えがあることがわかっているときに尋ねる。	95, 16.8	.859
追加	議論の中で抜け落ちていた情報を示す。	61, 10.8	.875
相談	自分では先を見通せないときに他の人の意見や知識を求める。	51, 9.0	.885
提案	何らかの筋書きを持っているときにそれを示す。	43, 7.6	.807
CVS	共有フォルダ (CVS) ヘソースを更新したときに送られるログ。	43, 7.6	1.000
報告	自分が何か行動したときにその結果を知らせる。	41, 7.2	.946
賛成	自分も同じ意見であるときにその立場を表明する。	31, 5.5	.806
意見	論理的な裏づけなしで自分の意見を述べる。	16, 2.8	.746
反対	自分の意見が異なるときにその立場を表明する。	15, 2.7	.852
説明	過去のメッセージがわかりにくいと思ったときに説明する。	12, 2.1	.795
依頼	他人にやってもらいたいことをお願いする。	15, 2.7	.793
広報	何かを多くの人に伝えたいときに知らせる。	11, 1.9	.955
その他	—	29, 5.1	—

種類のスレッドがあるのかを見るために、スレッドを類型に分類した。表 2はその集計である。一つのスレッドに含まれる平均メッセージ数とスレッドが終了するまでに要した平均時間を単位「時」で表わしている。ここでわかるのは、ほとんどのコミュニケーション (68.5%) は、「相談」、「質問」、「報告」、「広報」の 4つのスレッドで行われているということである。

そして、Newconfig プロジェクトでのコミュニケーションパターンを明らかにするために、4つのスレッドについてメッセージタイプの流れを図 1に遷移図として示す。各タイプの生起確率は (その類型からの遷移の数)/(総遷移数)、類型 A から B への条件付き遷移確率は (A であるときに B に遷移する遷移の数)/(A からの総遷移数) である。図中で、円で囲んであるものは「開始」「終了」以外は、メッセージ類型であり、その中の数値はそのタイプの生起確率である。また、矢印に付随する数値は、条件付き遷移確率である。そして、遷移表に現われた遷移のうち、生起確率が 0.05 以下、あるいは、生起確率に条件付き遷移確率を乗じた遷移確率が 0.05 以下の遷移は省略した。また、出現頻度の大きい「回答」などの類型が入ると、例えばランダムに遷移したとしても、遷移確率は大きくなるため、それぞれの遷移に対してランダムを仮定したときの期待度数からの観測度数のずれについてノンパラメトリックな χ^2 検定を行なった¹。しかし、 χ^2 検定で

有意なずれが証明されなかった遷移も、頻度の大きな類型には多く遷移するという意味でランダムである可能性があるだけであり遷移図には含めてある。

相談スレッドの議論の流れは、図 1(a) からわかるように、複雑である。「相談」で議論が始まるが、多義性が大きい場合「質問」「回答」を繰り返すことで議論を絞り、「提案」という形で次の行動にむすびつけるようになる。質問スレッドは、単純な構造をしている。しかしながら、単に「質問」→「回答」で終了することは少なく、「それは…ですが…はどうですか?」のように「回答」が「質問」を伴う場合が多い。報告スレッドは比較的わかりやすい構造をしている。自分の作業の結果を「報告」し、それに対して「質問」「回答」を繰り返し、終了する。また、広報スレッドはほぼ報告スレッドと同じ構造をしている。

この分析で重要なのは、「追加」という類型がどこにでも出現し、重要な役割を果たしているということである。議論のどの時点でも「情報が抜け落ちていることを知り、情報を追加する」ことで、常に可能なかぎり最大限の情報が動員されていることがわかる。また、「質問」「回答」も同じようにすべてに出現している。議論のどの時点でも議論にあいまいさがある場合、それを「質問」して明確にすることで、議論に誤解がないようにしていることがわかる。

¹条件付き遷移確率 $P(B|A)$ が生起確率 $P(B)$ に等しいという帰無仮説。

表 2: スレッドの類型

スレッド 類型	特徴	出現数, 頻度 %	平均メッ セージ数*
相談	どう考えたらいいのかわからないときに相談する。	20, 22.5	6.6 (3.4)
質問	情報を要求し, 知っている人が提供する。	19, 21.3	3.7 (1.8)
報告	自分の行動の結果を他の多くの人に伝える。	13, 14.6	3.5 (3.0)
広報	ほかの人に知らせなければならないことを伝達する。	9, 10.1	3.9 (4.6)
提案	他の人の同意が必要ときに他の人の意見を求める。	6, 6.7	5.7 (2.7)
合意	ミーティングの日程や出席など全員の都合を調整する。	4, 4.5	8.5 (5.2)
調整	プロジェクトの進行状況を調べ報告する。	4, 4.5	3.5 (1.9)
その他	—	14, 15.7	—

* () 内は標準偏差。

5 発見事実

5.1 行動指向性

行動前の議論が行動の結果の報告に比べて極端に少ないことが観察された。さらには、何の宣言もなしに行動し結果を報告することが頻繁にあったということである。例えば次のメッセージは、TODO リストにあった「最新のベータリリースへの更新」という作業の結果報告である (ただし、名前やメールアドレスなどのプライベートな情報は筆者によって変更が加えられている)。このメッセージの参加者はこれまで一度もプロジェクトのメンバーと接触したことはなく、メイリングリストやウェブページを見て参加したものである。

Date: Sep 8, 1998 20:39:38
To: newconfig@jp.freebsd.org
From: OM
Subject: [newconfig 176] 3.0-19980804-SNAP and newconfig
OM と申します。

newconfig を 3.0-19980804-SNAP にマージしてみました。

作業としては、
1. reject したのは 3 つくらいでした。適当に手でマージしました。
2. kern/kern.opt.c, i386/include/conf.h がなくなっていたので、newconfig の方からコピーしました。
3. make depend すると、opt.vmpage.h と opt.nfsroot.h がないとおこられるので、3.0-19980804-SNAP の GENERIC ファイルで config して得られたのをコピーしました。
というような感じでした。

3.0-19980804-SNAP の kernel source に当てるパッチを
<http://.../archive/newconfig-980804.diff.gz>
に置いておきました。

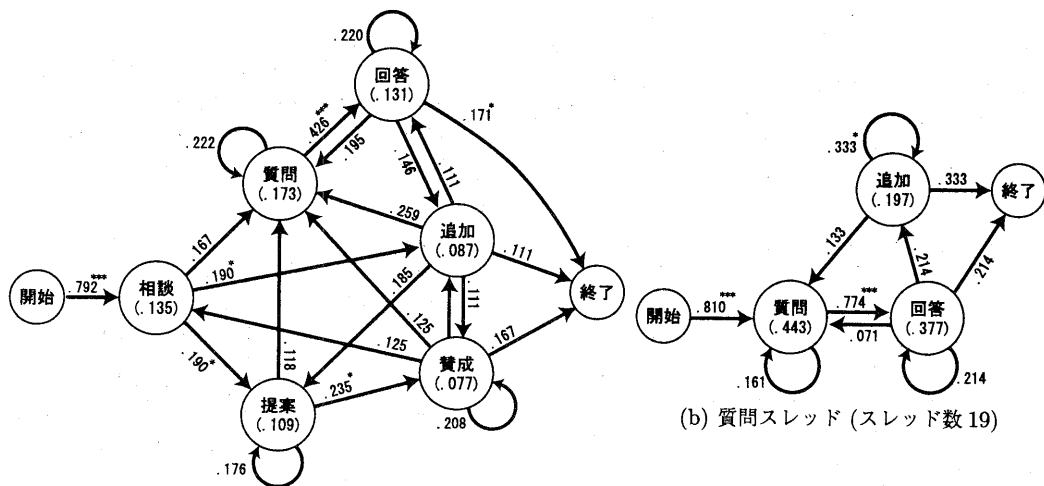
少なくとも syscons, wd は動いています。
上の 3. は、config の仕組みがまだよく分かっていないのでこのパッチでは対処できていません。

(略)

このような行動指向は、電子メディアのいくつかの制限に帰することができる。ひとつには、多義的なメッセージを伝達できない [3] ために、行動して結果を見た方が早いということである。ここで重要となるのは、作業の重複をできるだけ避けるために早い完成が必要であるということと、他人の援助なしに行動することができるという点であり、高い個人の能力が求められる。次に、電子メディアでは年齢、性別、肩書などの社会的文脈が抜け落ちるために、人々は自己中心的になるという傾向 [10] がよい方向に作用していると考えられる。通常では行動を起せない人でも電子メディア上では活動的になる。最後に、電子メディアは個人の社会的文脈であるアイデンティティを隠すために、個人は行動することでアイデンティティを確立しなければならない [4]。実際にプロジェクトでリーダーシップを取る人は、マネージャやプロジェクトリーダーといった肩書ではなく、行動を通して他の参加者から認められた人である。

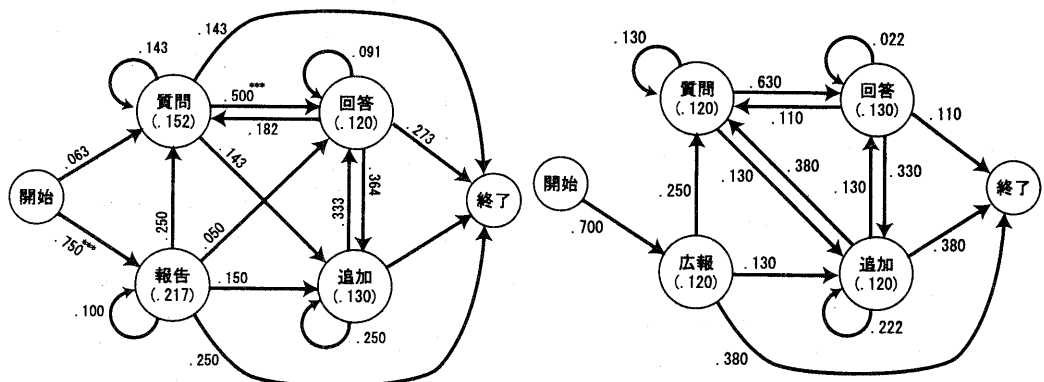
5.2 オーバラップ型開発

従来の開発では実装は製品の仕様の決定後に始まるが、オープンソース開発ではこの 2 つのフェーズが完全に並列する。企業での開発でも従来のウォーターフォール型に対するスパイラル型のように仕様の決定に段階を設ける手法がある。しかしながら、スパイラル型でも、各段階で仕様と中間目標を決定しそれを実装して、次の段階に入るという過程をたどるため、基本的には実装の詳細を決定してから実装に取りかかるのである。企業における開発は、ソフトウェア開発にはコストがかかり失敗は許されないという前提にあるために、計画が決定しない限り実装に取りか



(a) 相談スレッド (スレッド数 20)

(b) 質問スレッド (スレッド数 19)



(c) 報告スレッド (スレッド数 13)

(d) 広報スレッド (スレッド数 9)

(注) すべての図に関して、 $p < .05^*$, $p < .01^{**}$, $p < .001^{***}$

図 1: 各スレッドにおける議論の流れ

かれないというのが理由である。

オーバーラップすることは単に仕様の決定から製品のリリースまでのリードタイムを短縮させる [6] だけではない。重要なのは、仕様が行動と密接に関連しているという点である。言い換えれば、企業での管理された開発は事前に規定された「正しいと信じられている (espoused)」仕様に基いているが、オープンソース開発では「状況に埋め込まれた」仕様に基いている。正しいと信じられた仕様は、人間の限定合理性のために完全ではなく実際の実践から剥離していくが、状況に埋め込まれた仕様は複雑で一般化できないが、適切な行動を導く [1]。

さらに、電子メディアはアイデンティティを隠す

ために、オープンソースでは固定的な肩書きや権限などは存在しない。階層構造や肩書きなど効率を優先するために制定された (enacted) 構造は、新しい取り組みに対しては必ずしも有効に働かない [5]。むしろ、イノベーションの機会を減少させるのである。オープンソースでは固定された仕事分担や権限の割り当ては存在せず、行動のみが組織を形作っている。基本的に、電子メディアの限定性のゆえに、効率よりも柔軟性を優先するのである。

5.3 効率的な技術の結合

電子メディアは複雑な情報の伝達が困難でありや心理的な影響が弱いために、イノベーションのような

創造的なプロセスには適さない。しかしながら、確立された技術の結合は、むしろ電子メディアの方が効率的であることが発見された。内容分析でも、情報の結合に関与する「追加」と「質問」の2つの類型が議論の様々な段階で多く出現していた。次のメッセージはその例である(文中、略は筆者によるもの。また、プライベートは内容は置き換えてある)。pccardd や dynamic configuration manager や docking station といった専門用語によって複雑な議論がすすきり行われている。特に、別の技術を組み合わせることによって新しい技術を提案していることに注目しなければならない。

Date: Thu, 2 Jul 1998 17:16:36 +0900 (JST)
From: HN
To: newconfig@jp.freebsd.org
Subject: [newconfig 77] Re: openbsd bus separation

(略)
>あと、pccardd の問題もありますね。今は ed0 とか直接書いているので、
>これもまずい。だれか pccardd を kernel の中に突っ込んで、まともに
>して。:-)

今の段階で突っ込んだじゃうとデバッグしにくくなっちゃうような気も...

pccardd をもっと一般化した、dynamic configuration manager みたいのがほしいですね。docking station なんかも扱える。kernel 内につくりこんじゃえば、もはや静的な configuration と区別する必要はないともいえる?

この事実は電子メディアの積極的側面により説明できる。まず、電子メディアは情報の意味の伝達(多義性の解決)には適さないが、情報の量の伝達(不確実性の減少)には適している[3]。そして、電子メディアが非同期であるために、発言の機会が失なわれない。さらに、テキストによるコミュニケーションであるために、全員が同じ正確な情報を共有できる。また、電子メディアを介した関係は「弱連結」であり、よりリソースに到達しやすく、情報の収集が容易である[2]。

6 おわりに

本研究で得られた発見事実は、企業での情報システムの導入や社会の情報化などにおいても重要な示唆になると考えている。しかしながら、今回は単一ケースのみを扱い、研究の手法も観察、インタビュー、内容分析といった比較的定性的なものにとどまっている。今後はケースを広げるとともに、質問表調査などで定量的に検証していきたい。

参考文献

- [1] John Seely Brown and Paul Duguid. Organizational learning and communities-of-practice: Toward a unified view of working, learning, and innovation. *Organization Science*, 2(1), 1991.
- [2] David Constant, Lee Sproull, and Sara Kiesler. The kindness of strangers: The usefulness of electronic weak ties for technical advice. *Organization Science*, 7:119-135, 1996.
- [3] Richard L. Daft and Robert H. Lengel. Organizational information requirements, media richness and structural design. *Management Science*, 32(5):554-571, 1986.
- [4] Judith S. Donath. Identity and deception in the virtual community. In Marc Smith and Peter Kollock, editors, *Communities in Cyberspace*. Routledge, London, 1999.
- [5] Rebecca Henderson and Kim Clark. Architectural innovation: The reconfiguration of existing product technologies and the failure of established firms. *Administrative Science Quarterly*, 35(1):9-30, 1990.
- [6] Marco Iansiti and Alan MacCormack. Product development on the internet. In Stephen P. Bradley and Richard L. Nolan, editors, *Sense and Respond: Capturing Value in the Network Era*, pages 175-200. Harvard Business School Press, 1998.
- [7] Klaus Krippendorff. *Content Analysis: An Introduction to Its Methodology*. Sage Publications, 1980.
- [8] Robert H. Lengel Richard L. Daft and Linda K. Trevino. Message equivocality, media selection, and manager performance: Implications for information systems. *MIS Quarterly*, 11:353-364, 1987.
- [9] John Short, Ederyn Williams, and Bruce Christie. *The Social Psychology of Telecommunications*. John Wiley & Sons, 1976.
- [10] Lee S. Sproull and Sara Kiesler. Reducing social context cues: Electronic mail in organizational communication. *Management Science*, 32(11):1492-1512, 1986.