

Title	企業によるオープンソースソフトウェアへの特注 ： 見過ごされてきた課題の抽出
Author(s)	島田, 義秀
Citation	年次学術大会講演要旨集, 37: 441-445
Issue Date	2022-10-29
Type	Conference Paper
Text version	publisher
URL	<a href="http://hdl.handle.net/10119/18563">http://hdl.handle.net/10119/18563</a>
Rights	本著作物は研究・イノベーション学会の許可のもとに掲載するものです。This material is posted here with permission of the Japan Society for Research Policy and Innovation Management.
Description	一般講演要旨

## 企業によるオープンソースソフトウェアへの特注 ～見過ごされてきた課題の抽出～

○島田義秀（株式会社デジタルガレージ）

### 1. はじめに

昨今、企業によるオープンソースソフトウェア（以下 OSS と表記）の活用は、オープンイノベーションのよく知られた典型となっている。OSS とは、ソースコードが公開され、第三者が自由に参照、変更、配布ができるソフトウェアである。また、このソースコードは、有志の開発者によって組成されるコミュニティ（以下 OSS コミュニティと表記）により開発、改善が行われている。代表的な OSS に Linux があり、一般的にオペレーション・システムと呼ばれるソフトウェアで 2020 年までに 4, 249 名が開発に貢献している。[1]

企業は、特に新たな市場へ参加しようとする時、開発しようとする製品やサービス（以下、製品と表記）に必要な技術の知識が不足しており、マーケットシェアも大きく水を開けられていたりする場合に、OSS の活用が選択される。また、先端 IT 技術は OSS として開発される事が多くなってきており、企業は、OSS を活用する事で研究開発のコストを節約できるという側面もある。

しかし、企業が OSS を自社の製品に組み込もうとする時、OSS の成熟度や完成度の低さ[2]などの理由からそのままでは使えない事があり、その場合は、OSS コミュニティへの提案を通して機能の追加や改善などを行う特注の必要性が発生する。この OSS の特注を行う時、企業への外注ではおおよそ起こり得ない問題が発生する。

本報告では、OSS として開発されるブロックチェーン技術へ特注を行った事例で分析を行い、発生する問題を整理し、その発生原理について詳述する。

### 2. 先行研究と理論的背景

OSS の活用という観点では、Joel West and Scott Gallagher[3]が、企業が OSS を活用してオープンイノベーションを成功させるための 4 つの戦略とその課題について示している。しかし、前節で述べたように、企業が OSS を活用してオープンイノベーションを進める場合の多くは、OSS へ企業が必用とする機能の追加や改廃という OSS の特注が発生する。この OSS の特注という観点での研究はなされていない。

また、日本企業における OSS 採用への課題を示す研究[2]がある。この研究で取り扱っている課題は、企業が OSS を採用するに至るまでの課題であって、この研究においても OSS の特注については触れられていない。

また、ソフトウェア開発の外注という観点では、東・細谷・高橋が、外注管理業務を分類し、中でも外注計画、受入検査及び外注評価の重要性について示した研究[4]がある。しかし、外注先としてソフトウェアの請負開発を生業とする外注業者を想定しており、OSS コミュニティへの特注については触れられていない。

OSS の特注は、これまで企業が外注業者へ依頼して行っていた開発外注とは違う。特に、営利を目的とした外注業者と、ボランティアで構成される OSS コミュニティでは、組織の構成が違ってくる。この組織構成の違いに着目した研究も筆者の知る限りでは見つからない。

企業にとって OSS の特注が必用となった場合に、どの様なリスクが存在するのかを把握できないと、マネジメントが困難となり、そもそも OSS を使用するという選択ができない場合もある。従って、企業が OSS を特注するプロセスの中で発生する問題について整理し、その発生原理を解明することに本報告の意義があると考えられる。

### 3. 研究手法と事例選択

本研究では、外注業者への開発外注と OSS コミュニティへの特注のプロセスや、組織の特徴を比較

するアプローチでケーススタディを実施し、発生する問題点と、その原因の仮説の抽出を試みた。調査は、題材の企業が使用している OSS のコミュニティが使用しているメーリングリスト、掲示板システム、グループチャットから得られる情報、OSS へ特注する際の事業活動を観察して得られる情報、及び、筆者の実務経験から得た情報を活用した。

事例研究の題材には、株式会社 **Crypto Garage** (以下 **CG** と表記) が取り組むブロックチェーンを活用した事業開発の事例を選択した。**CG** は、ブロックチェーン技術を活用した金融ソリューションの開発に特化している企業である。使用しているブロックチェーンは **OSS** で開発されており、同社の製品は補完材として位置付けられる。ゆえに、**OSS** へ強く依存する事業を行う事例となる。

また、**Linux** や **Apache** のような代表的な **OSS** を開発するコミュニティは、ある一定以上のコミュニティの規模を持ち、責任を本質的に委託された中心的な開発者グループの存在があり、意思決定階層のような仕組みを持っているという特徴がある。[5]

**CG** が活用するブロックチェーン・ソフトウェアも上記で示した **OSS** コミュニティの特徴を持っている。これら二つの理由から、この事例を研究対象とした。

#### 4. 事例紹介と分析結果

事例分析の結果、**OSS** の特注時に発生する、5つの問題が導出された。

- ① 誰に依頼して良いかわからない
- ② 提案は受け入れられたが開発が頓挫
- ③ 提案した仕様が変更される
- ④ 完成時期が予測できない
- ⑤ 対応されない納品後の不具合

以下では、この5つの問題について、企業の **OSS** 特注のプロセスの時系列に従って、問題毎に、外注業者へ開発外注する場合と、**OSS** の特注の場合とのプロセスの比較、事例の紹介という順序で詳述していく。そして、最後に5つの問題の背後にある共通の原因についても説明する。

尚、紹介する事例については、守秘義務等の関係上、一部、具体的には記載できない事を付記しておく。

##### 4.1. 誰に依頼して良いかわからない

企業が新たな機能を製品に追加しようとした時に、その製品で使用しているソフトウェアにも機能の追加または改変が必要となる事がある。当該ソフトウェアの開発を外注業者へ開発外注していた場合、企業は外注業者の営業担当、もしくは、カスタマーサポートというソフトウェアに関する問い合わせを受け付けてくれる部署へ連絡する。外注業者が用意する、要件をヒアリングする担当者に相談し、契約することで必要な機能が追加または改変される。

**OSS** で同じように機能の追加や改変が必用となった場合、**OSS** への問合せは **OSS** コミュニティが定める方法 (メーリングリスト、グループチャット、掲示板など) を通して行われる。問合せの内容は、**OSS** コミュニティの全員が閲覧可能で、誰でも問合せへの応答を投稿する事が出来る。言い換えると、誰でも問合せする事は出来るが、返答をする担当者は存在しない。

筆者の事例では、**bitcoin**<sup>1</sup> を利用している企業へ市場調査を実施し、**bitcoin** 資産の詳細な保有状況が見える化できるサービスに需要がありそうだということを発見した。しかし、このサービスを開発するためには、**bitcoin** の操作を容易にするモジュールの追加が必要であった。新規サービス開発を迅速に進めるために、機能追加の可否について相談しようとしたが、問合せ先がメーリングリスト、グループチャットや掲示板からコミュニティ全体に通知する方法しか無かった。そこで、グループチャットへ相談を投稿したが、興味を持ったコミュニティのメンバーが返答してくれるが、特定の誰かが担当者として対応してくれる事はなかった。

##### 4.2. 提案は受け入れられたが開発が頓挫

前述したように、外注したソフトウェアへの機能追加は、その外注業者へ相談、依頼する事から始ま

---

<sup>1</sup> **OSS** として開発される peer to peer 電子マネーシステム. <https://bitcoin.org/en/>

る。企業は、当該ソフトウェアへの要求を外注業者の担当者へ伝え、その後、両社が合意した要件が記載された開発契約を締結する。外注業者は、契約に従い要件を満たす機能を開発、追加したソフトウェアを企業へ納品する。

OSS へ機能の追加や変更が必要な時は、OSS コミュニティへ提案することができる。提案は、前述した方法と同様に、メーリングリストなどのコミュニケーションツールを使って、投稿する事が出来る。投稿した提案について、OSS コミュニティの合意を得ることで開発が行われる。しかし、OSS コミュニティに賛同を得て開始された開発でも、完成せずに終了される事がある。

筆者の事例では、elements で<sup>2</sup>暗号化に用いる鍵情報の導出方法の共通仕様が策定し実装する事を提案した。共通の仕様を定める事で、鍵情報のバックアップが簡素化され、また、異なるソフトウェア間での鍵情報の移行が容易になるという利点があった。そのため、この案には、elements の OSS コミュニティのコア・メンバー含む複数のメンバーが同意した。しかし、その機能の開発が進んでいる様子は見られない。

#### 4.3. 提案した仕様に変更される

企業は、ソフトウェアを外注し、取り決めた要件を満たす機能が追加されたソフトウェアであることを確認した後に受領し対価を支払う。当該ソフトウェアが提示した要件を満たして居ない場合は受領しない。こうして必要となるソフトウェアを調達している。

前述した事例の様に、OSS への機能の追加については OSS コミュニティの合意を得ることで開発が行われる。しかし、提示した要件通りに開発されない事がある。例えば、提案した要件をさらに良くしようという活動が行われることによって要件の変更が行われ OSS としてはより良くなるが、企業が想定していた要件との差異が生まれ計画の変更を余儀なくされる。

筆者の事例では、bitcoin を受け取る「新たな条件」<sup>3</sup>が追加されるという事が合意されていた。その条件を使用すると lightning network<sup>4</sup>が抱える複雑性の問題が解決する。そこで新たなサービスを直ちに企画したが、数ヶ月後に、既に合意していた条件をさらに拡張した条件に変更された。

#### 4.4. 完成時期が予測できない

企業にとって、計画を作ることは重要である。特に製品開発においては、その必要性は高い。その計画を基に、事業計画が作成されるからである。製品やサービスでソフトウェアが必要な場合、その開発を行う。自社で開発する部分と、外部から調達する部分を決定し、自社開発部分については開発スケジュールを作成する。外部から調達するソフトウェアについては、調達先となる外注業者から提示される費用と納期を参照する。この時、外注業者にソフトウェアの開発が必要な場合でも、ある程度確かな納期が提示される。企業は、外注するソフトウェアの納期を考慮したソフトウェア開発スケジュールを作成し、製品開発計画に組み込む事で、事業計画の策定が可能となる。

OSS では、OSS コミュニティに指揮命令系統は無く、コミュニティメンバーがボランティア活動としてソフトウェア開発を行っている。つまり、機能の開発は、コミュニティメンバー個人が計画し、または、計画無しに実行する。コミュニティメンバーが、個人の空き時間を使って開発をするという場合は、もちろん、納期などわからない。もし、計画が作成されたとしても、その計画を守らせる権利は本人を除いて無い。また、開発に取り組んでいることを公表せず、作業を進め完成したらパッチを公開する開発者も存在し、どれだけの開発者が、開発に取り組んでいるのかわからないという事もある。ゆえに、提案されている機能の開発スケジュールどころか、予測も難しく、ソフトウェア開発スケジュールを作成することが困難である。

筆者の事例では、bitcoin が決済完了性<sup>5</sup>を持たない事から派生する問題を解決する為のサービスである SETTLENET の企画を検討していた時、elements で構成されるブロックチェーンネットワークである Liquid Network<sup>6</sup>を決済インフラとして採用するとした。企画時点では機能の不足を認識しており、

<sup>2</sup> bitcoin のサイドチェーンとして開発されている OSS. <https://elementsproject.org/>

<sup>3</sup> bitcoin は、資産を送信する時に、それを受け取る為の条件を指定する事ができる。その条件は予めプログラムとして定義されている。

<sup>4</sup> bitcoin の決済を bitcoin の network を使わずに実行する事ができる技術である。

<sup>5</sup> 決済が無条件かつ取消不能となり、最終的に完了した状態

<sup>6</sup> elements を使用した、世界中の仮想通貨取引所、マーケットメーカー、ブローカーや金融事業者を繋ぐ決済ネットワーク

それらの開発と同時並行的に自社の企画を進めた。しかし、どの程度開発が進んでいるのか把握することが困難であった。また、開発が完了する時期についての予測も困難であった。

#### 4.5. 対応されない納品後の不具合

外注により開発されたソフトウェアは、企業の製品やサービスに組み込まれ顧客へ提供される。製品に何らかの問題が発生した場合は、その問題を迅速に解消する事が求められる。企業はソフトウェアの開発元企業と、ソフトウェア保守契約を結ぶことで、外注したソフトウェアの品質を担保している。一般的にソフトウェア保守契約では、問題が発生した時に問い合わせ先となる担当者が割り当てられる。担当者は、開発したソフトウェアとのすり合わせ部分に生じた問題に関する調査や、ソフトウェア自体に問題が発生した場合の修正パッチ開発などのサポートを提供する。

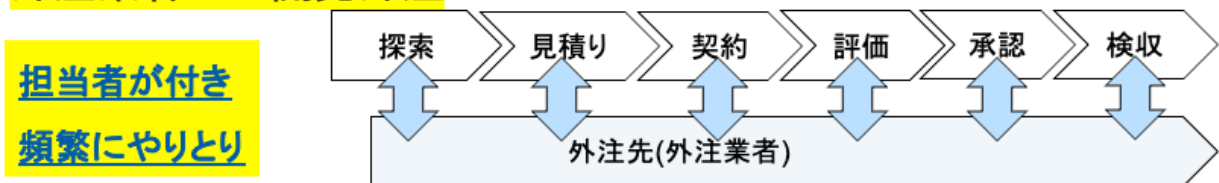
OSSを企業の製品やサービスに使用した場合にも、同様に問題は起こり得る。しかし、OSSを開発するコミュニティは、ボランティアで構成されており、主体が存在し無いため、保守契約を結ぶ事ができない。そのため、問題が起きた場合、前述した様に、不具合のレポートをコミュニケーションツールへ投稿することになる。しかし、すぐに返答が得られるとは限らないし、調査が始まるとも限らない。なぜなら、これらの作業もコミュニティメンバーのボランティア活動として行われているため、強制する事が出来ない、また、作業の進捗も把握が難しい。また、不具合の調査や修正パッチ開発においても、機能の改廃と同様のプロセスで実行されるため、少なくないコミュニティメンバーの作業が必用となる。さらに、ソフトウェアがある程度大きくなると、多くのメンバーが関わる事になり、メンバーも自分の責任で修正すべきかの判断が難しくなる。

筆者の事例では、暗号資産取引の決済サービスであるSETTLENETがbitcoinとelementsを使用しており、2020年6月にサービスを開始している。サービス開始時のソフトウェアのバージョンはbitcoinがv0.20.0、elementsがv0.18.1.7であった。サービス開始後にも、当該バージョンの不具合が、コミュニティが使用している掲示板サイトに投稿され、そのほとんどで対話がされているが、解決に至っていないものが残っている。

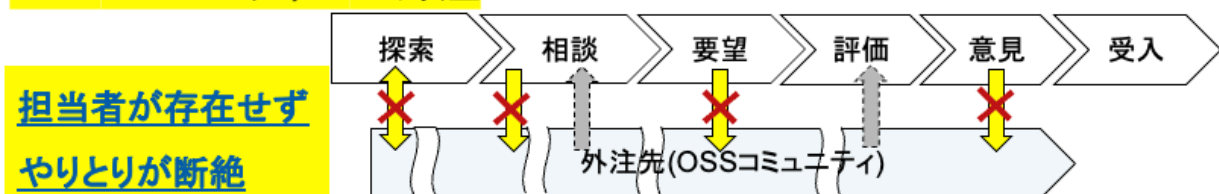
#### 4.6. 5つの問題の背後にある共通の原因

ここまで企業がOSSの特注をしようとする際に、外注業者へ開発外注する時には起こり得ない5つの問題について紹介した。その原因の一つとしてソフトウェアベンダーへの外注とOSSの特注とのプロセスの違い（下図参照）があると考えられる。

##### ■ 外注業者への開発外注



##### ■ OSSコミュニティへの外注



外注業者への開発外注では、プロセスの初期の頃から担当者が割り当てられ、最終的にソフトウェアが納品されるまでの間、開発の進捗や、機能の詳細な設計など、頻繁にやりとりをしながら進められる。

一方、OSSの特注プロセスでは、全体を通して担当者が存在せず、やりとりが断絶しながら進められる。例えば、新たな機能はメーリングリストへ投稿することで提案できるが、誰かが返答をしてくれるまで待つことしか出来ない。また、提案についての相談も特定の人物とやり取りするわけでは無く、ま

た受け入れられたとしても、実際に開発するのは、興味を持ったコミュニティメンバーである事が多い。そのため、提案を承認したコミュニティメンバーとは別となり、承認までの議論が引き継がれるわけではない。また、開発は複数人が協力して進める場合もあるが、大抵は個々に同時並列的に進めるため、開発の進捗を把握する事が難しい。さらには、知らぬ間に開発が頓挫していたりする。順調に開発が進められたとしても、コミュニティメンバーが行う仕様の変更についてパッチが完成するまで発見することは困難になる。

## 5. まとめと今後の課題

本発表では、事例から企業が OSS を特化するプロセスで発生する、企業へ外注した時にはおおよそ起こり得ない問題について整理を行った。さらに、その分析から 5 つの問題に共通する原因となる仮説の構築を試みた。

しかし、この原因は表面的であり、この原因が発生する原理については、まだ分析できていない。そこで、今後の課題として 2 点上げたい。

最初の事例である、機能の追加の依頼先が不明となる問題では、外注業者が担当者を割り当てるのは、最終的に開発を受注し売り上げを獲得するという動機に基づく行動である。他方、OSS コミュニティのメーリングリストへ提案を投稿しても返答してもらえないかは、その提案がコミュニティメンバーへの何らかの動機付けとなるかによって決まる。この様に、外注業者と OSS コミュニティのインセンティブシステムの違いによって生まれる問題は、他の事例で発生する問題の原因にもなっている様に考えられる。言い換えると、根本的原因は外注業者と OSS コミュニティの組織構造やマネジメントの違いにあると考えられる。そのため、組織論的な分類枠組みから比較分析を行いさらに分析を進める必要があると考えている。

もう一つの視点として、依頼関係の構築方法の違いも深層の原因として存在すると考えられる。外注業者へ開発外注する場合は、契約という形でソフトウェア開発の依頼と請負という関係が成立する。他方、OSS では契約という事ができない、ゆえに、依頼し合意した内容でも実行の責任が無く、実行されなかった時に処罰を与える事も出来ない。この依頼関係の構築がどのような影響を与えているのかについては、今後、さらに調査研究を行なっていく必要があると考えている。

## 参考文献

- [1] 2020 年度版 Linux カーネルの歴史レポート, The Linux Foundation (電子書籍版) (2020)
- [2] 三浦和夫、石井幸太郎, 日本企業の OSS 採用への課題と将来展望, UNISYS TECHNOLOGY REVIEW 第 94 号 (2007)
- [3] Joel West and Scott Gallagher, Patterns of Open Innovation in Open Source Software, Open Innovation: Researching a New Paradigm, Oxford University Press (2006)
- [4] 東基衛, 細谷僚一, 高橋宗雄, ソフトウェアマネジメント: ソフトウェアマネジメント概説, 情報処理 33 卷 8 号 (1992)
- [5] Steven Weber, The Success of Open Source, Harvard University Press (2004)