

オープンソースによるビジネスに関する検討

A Study on Open Source Business

前田和昭[†]

Kazuaki Maeda[†]

[†] 中部大学 経営情報学部

[†] College of Business Administration and Information Science, Chubu Univ.

要旨

本稿では、開発者の視点とビジネスの視点から、オープンソースについて検討する。オープンソースの良さは、無料で利用できるところにあるのではない。無料公開することで利用者と開発者を増やし、開発とビジネスのスピードを加速するところに良さがある。しかし、日本からのオープンソースに対する貢献は、非常に小さいのが現状である。そこで、これまでの経験を踏まえ、また、これから筆者がオープンソースで活動する過程を考慮に入れながら、開発者の視点とビジネスの視点から議論する。

1. はじめに

一般に使われている商用ソフトウェアでは、オブジェクトコードのコピーのみが利用者に渡されている。ソースコードを利用者に渡すことは、特別な契約を結ぶ場合以外にはないと言われてきた。企業が高価な人件費をかけて作り上げたソースコードの中には、社外には漏らしたくない重要なアイデアや、生産性を上げるために長年に渡って苦労して開発したプログラム部品が含まれる。競合する他社に対して優位性を維持するためには、ソースコードを隠しておくのが当然とされてきた。

ソフトウェアの世界では、新しいアイデアが生まれてから約20年が経過してようやく広く一般に利用されると言われている。オープンソースソフトウェア（以後OSSと略記する）は、プログラムのソースコードを公開し、かつ、利用者と開発者とビジネスのそれぞれに何らかの恩恵を与えるものだと考えることができる。このソースコード公開に大きな影響を与えたのは、1984年に始まったGNUプロジェクト[1]であり、それから20年以上が経過した。また、Linuxに代表されるOSSの開発形態がバザール方式と言われるようになってから約10年が経つ[2]。現在では、既にOSSが定着してきたように思える。ここでは、ソフトウェアビジネスを根本的に変えてしまったOSSについて、ビジネスの視点と開発者の視点から現状を述べることにする。

2. オープンソースソフトウェアの意義とビジネス

2.1. ソースコード公開の意義

ソースコードは人間が解読可能な表現で記述されているため、ソースコードがあればプログラム内部の秘密を解き明かすことができる。このソースコードをプログラム開発者が入手すると、以下の5つのことが可能になる。

第一に、教育用の良い資料としてソースコードを閲覧することが可能になる。熟練したプログラム開発者が記述したソースコードには、貴重なノウハウがたくさん詰まっている。そのため、昔からソースコードはプログラミングの教科書であると言われ、実力を向上させるために重要な役割を果たしてきた。

第二に、ソースコードを公開することで、不足した機能を追加することが可能になる。プログラム開発者は、不便なことを改善して自分の生産性を上げるために、プログラムを開発することが多い。もし、他人が開発した便利なプログラムをソースコードと一緒に入手できたとすると、自分にとって足りない機能があれば、その機能を自分で追加して、さらに使い続けることがある。

第三に、活動が終了したプログラムを復活させることが可能になる。プログラム開発が特定の企業で行われた場合、その製品がどんなに優れていても、ビジネス上の決定によって開発が終了することがある。例えば、インタビュー記事[3]によれば、Cloud Engines社のCEOが、以下のように発言している。

(該当記事[3]から抜粋)

私は会社を設立した当初から宣言していることがある。それはもし *Cloud Engines* が倒産した場合、*Pogoplug* に関する技術をすべてオープンソース化することだ。そのため *Cloud Engines* としてサービスを継続できなくても、オープンソースを利用して第三者がサービスを作り利用することが可能になる。

このように、もし開発が終了して活動が止まってしまっても、そのプログラムのソースコードが公開されれば、他のグループによってプログラム開発を続行させることができる。

第四に、別のプラットフォームに対応することが可能になる。プログラムを開発するときは、ある特定のプラットフォーム（例えば *Windows*）上で開発とテストを行う。開発の途中もしくは開発した後に、そのプログラムを他のプラットフォーム（例えば *Linux*）に移植することで複数のプラットフォームに対応したプログラムができあがる。

第五に、自分で障害に対応することが可能になる。商用で販売されている製品であっても、時々正常とは思えない動作をすることがある。商用のプログラムのソースコードは非公開となっているため、どんなにコンピュータに詳しいプログラム開発者であったとしても、その障害を取り除くことは極めて難しい。しかし、ソースコードが公開されている場合、知識と経験豊富なプログラム開発者であれば、その原因を調べ、ソースコードを修正することで障害を取り除くことができる。

2.2. オープンソースによるビジネス

以前から大学や研究機関では、研究成果として、開発したプログラムをソースコードと一緒に公開することが当然のように行われていた。ところが、商用ソフトウェアが普及するにしたがって、ソースコードを未公開とすることが常識となり、昔から大学研究機関で行われてきた知識の共有が難しくなった。それに反発し自由な社会に戻すために、1984年に *Richard Stallman* が立ち上げたのが *GNU* プロジェクトである[1]。GNUプロジェクトで開発されるプログラムは、自由なソフトウェアを目指すことから「フリーソフトウェア」と呼ばれている。フリーソフトウェアには、利用者が自由にプログラムを実行し、ソースコードを自由に読んでソースコードを自由に変更し、ソースコードの変更前/変更後の両方を配布/入手する権利が与えられている。この自由な思想を法的に保護するために *GPL* (*GNU General Public License*) が作られた。しかし、*GPL* は、ソースコードを公開するための自由を尊重する思想が強すぎたため、フリーソフトウェアをビジネスで活用することが難しい場合が多かった。

1991年にヘルシンキ大学の大学生 *Linus Torvalds* が自分で開発したオペレーティングシステムをインターネット上で公開した。これが *Linux* の始まりである[4]。Linuxはソースコードを含めて公開され、インターネット上のプログラム開発者たちはそのソースコードを見ながら、足りない機能を次々に追加していった。公開されたプログラムはインターネットを介して多くのプログラム開発者たちに渡り、プログラム開発者たちの更なる貢献によって、不完全な部分の完成度が上げられた。このサイクルを速く多く繰り返すことでいつの間にかビジネスで十分使える信頼性を持つレベルにまで成長した。学生が趣味で始めた *Linux* は、今では多くの企業においてビジネスの中心的役割を演じるようになっている。

企業が高い費用をかけて開発したプログラムを、OSSとして無料公開してしまうと、損失ばかりのように思える。しかし、OSSの活動に参加している企業は、利益を出すためにOSS活動を進めている。OSSに対するビジネスへのアプローチはいろいろある。

例えば、誰でもが無料で自由に使えるソフトウェアをOSSで公開し、それを活用する場面でビジネスを生み出すアプローチがある。IBMが開発した *Eclipse*[5]がその典型例である。自社の開発者が中心となってOSSを開発し、そのOSSが世間に広く浸透すれば、自社がそのOSS内部に一番精通した会社となる。したがって、世間に広まったOSSを土台にシステムを開発する案件が発生し、また、出版物やセミナーなどからも収入を得ることができる。

さらには、開発したプログラムをOSSとして無料公開すると、なるべく多くの開発者の手に渡ることとなる。*Eclipse*が開発者に急激に広まったのも、無料公開されていたことによるところが大きい。もともとはJava言語でのプログラム開発を支援するために作られたものだが、プログラム部品をプラグイ

ンとして組み込むことができるため、いろいろな開発環境（例えば、C/C++ IDE や UML ツール）が構築され公開されている。

また別のアプローチとして、ソフトウェア開発のときには無料で利用できるものの、それを組み込んだ製品を出荷するときに利用料金を発生させる方法がある。代表的なものが MySQL[6]であり、デュアルライセンスと呼ばれている。MySQL に親しみを感じる開発者を増やすことで、MySQL を土台としたプログラムが増え、ビジネスの機会が増える。また、ソースコードが公開されているため、何か問題が発生したときに、開発者がその問題に対処しやすい。

3. オープンソースによるビジネスへ向けて

製品を OSS として無料で配布し、その複製も許可するスタイルでは、ビジネスが成立しないよう思える。しかし、Linux や MySQL などの成功例があり、無料で提供する製品とそこから派生するサービスをうまく展開すれば、ビジネス化の可能性は十分にあり得る。製品を無料で配布することで世界中に広く浸透させ、それを土台にビジネスを確立できるはずである。また、開発者魂を鼓舞するようなソフトウェアを OSS として公開し、外部の開発者に先進的な開発を委ねることで、開発のスピードを向上させることができるはずである。

そこで、筆者が独自に研究開発してきた成果を、本格的なデータ処理アプリケーションとして広く利用可能なものへと発展させ、OSS によるビジネスとして展開することを考えている。本節では、筆者の研究成果とそれをビジネスに結びつけていくための今後の予定について述べる。

3.1. 新しいデータ記述方式 RugsOn

データ処理アプリケーションの多様な場面で、XML をデータ記述形式として使うことが多くなってきた。しかし、最新のプログラミング言語が XML データの処理に向いているわけではない。一方、最近の Web アプリケーションでは、データ記述に JSON を利用することが増えてきた。JSON は JavaScript のサブセットを使っているため、データ記述とプログラム記述が同じ言語となり、整合性が高くなる。しかし、JavaScript 以外のプログラミング言語を使った場合は、XML の場合と同様、データ記述とプログラム記述が別の言語を使うことになってしまう。

そこで筆者は、最新のプログラミング言語で構造化データを扱うために適したドメイン特化言語（Domain Specific Language、以後 DSL と略記する）として RugsOn を設計し、RugsOn で記述された構造化データを処理するためのソフトウェアツールを開発し続けている。

DSL とは、特定のドメインに特化してカスタマイズした言語のことである[7]。適切に設計された DSL があれば、効率的なシステム開発が可能になる。例えば、プログラミング言語 Ruby は、内部（internal）DSL のためのホスト言語として重要な役割を果たしている。内部 DSL とは、ホスト言語の適当なサブセットを使う DSL である。内部 DSL の場合、ホスト言語の処理系をそのまま利用できるため、書かれたテキストを読み込むためのパーザを作成する必要がない。Ruby をホスト言語とする内部 DSL であれば、Ruby のメソッド定義を必要なだけそろえることで、DSL で書かれたテキストが実行可能になる。

通常のプログラミング言語では、メソッド呼び出しを記述するときに実引数を括弧で囲むことが多い。それに対して、Ruby では、メソッド呼び出しの括弧を付加しても良いし省略しても構わない。括弧を省略すると、表現が英語的になり読みやすくなるのが、Ruby を内部 DSL のホスト言語にする利点の一つと言われている[8]。また、複数の文をまとめるためのブロックがあり、メソッドの引数として効果的に使うことで、記述能力が格段に高くなる。RugsOn は、Ruby, Groovy, Scala に共通な構文を使って構造化データを表現するための内部 DSL である。ブロックで親子関係を表現し、括弧を省略したメソッド呼び出しを使うことで構造化データを記述する。

3.2. ソフトウェア公開によるビジネス

データ処理アプリケーション開発における OSS でビジネスを立ち上げるために、

- 開発支援ツールを OSS として世界中に配布
- 既存システムを最大限利用して試作する RugsOn 専用のシステムを開発
- OSS で利益を獲得するためのビジネスモデルの確立
- 世界中のソフトウェア開発者に、これらの開発成果を公開しビジネスでの利用を実現

を予定している。これらを世界中のソフトウェア開発者に公開し、共同作業によって推進させたい。

データ処理アプリケーション開発で OSS を土台にしてビジネスを立ち上げるには、高性能の処理を実現できることを強く主張する必要がある。これを短期間で行うために、既存システムを最大限に利用し、RugsOn によるデータ処理アプリケーションが性能面で既存システムに劣らないことを実証する。開発するシステムを独自のものとし、ライセンスを自由に付与することがビジネスを立ち上げるためには非常に重要である。そこで、既存のシステムに依存しないように、システム全体の再開発を進めることも考える。

IT 産業においては、ソフトウェアを開発する企業の活躍は重要な位置を占める。日本企業へのアンケート調査の結果[9]からも明らかなように、日本におけるソフトウェアの輸出額と輸入額の間には、30 倍近い極端な格差がある。この格差を縮めるには、従来と同じ手法では通用するはずがない。日本国内でのソフトウェア開発を活性化するために、国際的な視点でソフトウェアビジネスの活性化が必要と思われる。

そこで、世界中のソフトウェア開発者に RugsOn を中心としたソフトウェアを OSS として公開するために、筆者自身が広告塔となって、日本国内外を問わずオープンソースに関する会議にて、講演者として参加することを目指している。現時点（2011 年 11 月上旬）で、11 月中旬にオーストラリアで開催される会議 OSDC (Open Source Developers Conference) [10]で講演することが予定されているので、これを OSS によるビジネスを立ち上げるための広報活動の手始めとする。さらには、OSS で世界的に著名な開発者会議として、ヨーロッパで開催される FOSDEM[11]と、米国で開催される OSCON (2011 年に開催された開発者会議の Web サイト[12]) での講演を目指し広報活動の範囲を広げる。今後、優秀な技術であることを浸透させることと、OSS として配布できる優れた製品を開発することに注力する予定である。

謝辞

本研究は、独立行政法人科学技術振興機構 研究成果展開事業 「研究成果最適展開支援プログラム (A-STEP) フィージビリティスタディ【FS】ステージ 探索タイプ」の支援を受けている。

参考文献

- [1] GNU プロジェクトの思想, <http://www.gnu.org/philosophy/philosophy.ja.html> (参照 Nov. 1, 2011).
- [2] Raymond, E. S., 伽藍とバザール, 光芒社, 2001.
- [3] 【PC Watch】 Cloud Engines、パーソナルクラウドデバイス「Pogoplug」を国内販売, http://pc.watch.impress.co.jp/docs/news/20110202_424412.html (参照 Nov. 1, 2011).
- [4] Torvalds, L. and Diamond, D., それが僕には楽しかったから, 小学館プロダクション, 2001.
- [5] Eclipse -The Eclipse Foundation open source community website, <http://www.eclipse.org/> (参照 Nov. 1, 2011).
- [6] MySQL :: The world's most popular open source database, <http://www.mysql.com/> (参照 Nov. 1, 2011).
- [7] Fowler, M., Domain-Specific Languages, Addison-Wesley, 2010.
- [8] 角谷信太郎, Ruby on Rails 究極指南 Ruby on Rails 究極指南: 第 4 回 話題騒然! 「言語内 DSL」の概念と Rake, <http://www.ibm.com/developerworks/jp/opensource/library/itm-rails4-rake2/> (参照 Nov. 1, 2011).
- [9] 中小企業総合事業団,平成 13 年度ベンチャー関連情報収集・提供・調査事業ベンチャー 企業に関する国内外の直接金融(投資)環境状況調査報告書, 2002.
- [10] OSDC 2011 | Nov 14-18 Canberra, <http://osdc.com.au/> (参照 Nov. 1, 2011).
- [11] fosdem.org, <http://fosdem.org/2012/> (参照 Nov. 1, 2011).
- [12] OSCON 2011 - O'Reilly Conferences, July 25 - 29, 2011, Portland, OR, <http://www.oscon.com/> (参照 Nov. 1, 2011).