

論 説

オープンソース・ソフトウェアと
ビジネスとの関係に関する考察

竹 田 昌 弘

目 次

1. はじめに
2. オープンソース・ソフトウェアとその開発コミュニティ
 - (1) フリーソフトウェアの誕生
 - (2) オープンソース・ソフトウェアと Linux
 - (3) オープンソース・ソフトウェア開発コミュニティ
3. オープンソース・ソフトウェアとビジネスとの関わり
 - (1) 寄生モデル
 - (2) 共生モデル
 - (3) 一体化モデル
4. オープンソース・ソフトウェアを巡る伝統的ビジネス（寄生モデル）
 - (1) ソフトウェアの配布
 - (2) コンサルテーション
 - (3) オープンソース・ソフトウェアをもとにした受託開発
 - (4) 運用支援
 - (5) 保守
5. オープンソース化される商用ソフトウェア（1）...（ビジネスとの競合）
 - (1) Netscape Navigator の歩み
 - (2) Microsoft Internet Explorer との競争
 - (3) オープンソース化
 - (4) 現在の Mozilla
6. 戦略的地位構築手段としてのオープンソース・ソフトウェア（共生モデル）
 - (1) Linux との関わり
 - (2) 相対的優位性の獲得
7. オープンソース化される商用ソフトウェア（2）...（一体化モデル）
 - (1) ニュートキーヨーのセルベッサ
 - (2) オープンソース・ソフトウェアによる企業間ネットワークの形成
8. まとめ

1. はじめに

Linux オペレーティングシステムの普及によって注目されているオープンソース・ソフトウェアとは、ソフトウェアの設計図ともいえるソースコード¹⁾を公開して、配布し、誰でもが

1) コンピュータで稼働するソフトウェアは、コンピュータが直接解釈・実行できる 2 進数化されたプログラム
(次頁に続く)

自由に利用し、開発できるソフトウェアである。オープンソース・ソフトウェアの開発は、自発的に参加する開発者たちによる活動が中心である。オープンソース・ソフトウェアの一部は、市場に流通する商用ソフトウェアに匹敵する機能と品質を備え、競合状況も発生している。しかし、オープンソース・ソフトウェアは、その周辺でのビジネスを拒絶することなく、ビジネスとの間に一定の共存関係を保っている。

最近では、オープンソース・ソフトウェアの周辺でのビジネスだけでなく、企業戦略の中に深く取り込まれたり、企業間ネットワークを形成する手段となったり、より複雑な相互作用も見られるようになり、二者間の関係は進化を続けている。これは、情報産業のビジネスの中心が、ハードウェアから、ソフトウェアに、そしてさらにソリューションに移行している現実とも大きく関係している。

本稿では、まず、オープンソース・ソフトウェア開発の特徴を概観する。その上で、ビジネス・コミュニティとの間の相互関係について、基本的な共存関係から、のちにビジネス・コミュニティが自らの経営戦略の一部としてオープンソース・ソフトウェアを取り込んでいくまでについて、事例をまじえて整理する。

2. オープンソース・ソフトウェアとその開発コミュニティ

オープンソース・ソフトウェアの歴史は、ソフトウェアを巡るビジネスへの反発から始まっている。

コンピュータ産業の初期、ソフトウェアはハードウェア製品の付属品としての位置づけでしかなかった。ハードウェア製品同士の互換性は低く、したがって付属するソフトウェアの流通範囲は狭いので、ソフトウェアはビジネスになりえなかった。ハードウェア製品の価値を高めるため、ソフトウェアの設計図であるソースコードを公開し、先進的な利用者たちが自主的に開発・改善したソフトウェアの相互交換を許容していた。すなわち、利用者コミュニティのレベルがハードウェア製品の価値を決めていたといってもよい。

IBM のシステム 360 発表以降、ハードウェア製品が互換性の高いファミリマシンとして開発され、ビジネスでのコンピュータ利用が普及してくると、ソフトウェアは単独で商品と見なされるようになり、保護が強化されるようになる。なぜなら、ファミリマシンの登場によって、一度開発したソフトウェアはファミリ内のすべてのコンピュータ製品で動作し、また設置台数

であるが、これを直接記述することはない。ソフトウェアの開発では、どのようなソフトウェアを作成するのかを決めた(仕様決定)のちに、その実現方法を階層的に詳細化(詳細設計)し、人間が理解しやすく記述しやすいプログラム言語を用いて記述する。このプログラム言語で記述されたプログラムをソースコード(あるいはソースプログラム)という。ソースコードは、コンパイラなどの翻訳ソフトウェアによって、個別のコンピュータにあわせて2進化されたプログラムに変換され、実行される。

の増加もあり、ソフトウェアがビジネスとして成立するようになったからである。この結果、ソフトウェアのソースコードは開示されず、ソフトウェアの中身を調べることは禁じられるようになった。

(1) フリーソフトウェアの誕生

MITの人工知能研究所に勤務していたRichard Stallmanは、このようにしてソフトウェアの自由が奪われていく状況に反対するため、1984年にGNUプロジェクト²⁾を開始するためにMITを退職³⁾し、1986年にフリーソフトウェア財団(Free Software Foundation)を設立した。Stallmanは、ソフトウェアを人類共有の財産として自由に利用することで、さらに優れたソフトウェアの開発につながると考えている⁴⁾。

Stallmanが提唱するフリーソフトウェアの「フリー」は自由を意味し、ソフトウェアの実行、複製、配布、調査、変更、改善に関する自由の保証を問題にしている。この自由を確保するために作られたのが、GNU GPL (General Public License) というライセンス文書⁵⁾である。ソフトウェアの調査、変更、改善の自由を保障するため、GNU GPLではソフトウェアの配布にあたって、ソースコードを含めて配布することを定めている。

GNU GPLでは、ソフトウェアの自由を保障するための条項を定めているが、そこではソフトウェアを使ってビジネスを行うことを排除してはいない。自由を享受するために利用者を支援する活動から利益を得ることは、積極的に認められている。実際のところ、フリーソフトウェア財団は運営経費を捻出するために、ソフトウェアの配布CDROMを財団から直接購入することを求めている。

(2) オープンソース・ソフトウェアとLinux

自由を求めるフリーソフトウェアの活動から派生して、ソースコードの公開を保証しながらソフトウェアの開発を進めていくプロセスに注目したグループが出現した。彼らは、ソフトウェ

2) GNU: Gnu is Not Unix ソースコードの管理強化が始まったUnixオペレーティングシステムへの反発から命名している。GNUプロジェクトでは、自由に利用できるGNUオペレーティングシステムと、その周辺のさまざまなソフトウェアの開発を目的としている。

3) MIT勤務者の開発するソフトウェアの権利はMITに帰属するという規定のため、MITに勤務したままではソフトウェアの自由が守れないと判断し、StallmanはMITを退職をした。

4) この考え方は、学術研究の考え方に近い。研究成果は公表され、過去の成果をもとにして新たな成果を作り出していく。このための参照や引用は原著を明示したうえで一定の範囲で認められる。

5) ソフトウェアの著作権を放棄するだけでは、そのソフトウェアをもとにして独占的なソフトウェアを作られる恐れがある。したがって、永続的に自由に利用されることを保証するためには、著作権に基づくライセンスによってこれを定めなければならない。

アの自由を求めるといふ思想的運動よりも、新しいソフトウェア開発プロセスを可能にする機能に注目し、オープンソース・ソフトウェアという名称を作り出した。

Linux オペレーティングシステム⁶⁾ は Linus Torvalds が開発を始めたものであるが、GNU GPL を適用してソースコードの公開を保証する形で開発が継続されている。しかし、その動機付けはソフトウェアの自由を通じた人類の財産作りなどではなく、Torvalds が「それが僕には楽しかったから」⁷⁾ ということに代表されるように、個人的な興味あるいは利益追求のためであった⁸⁾。

(3) オープンソース・ソフトウェア開発コミュニティ

Linux 以降、オープンソース・ソフトウェア開発の一般的なモデルが構成されていった。その中心となっているのはインターネットの普及である。それまでは遠隔で共同作業をするためのファイル交換は物理的に磁気テープ媒体を送りあうか、数日かけてパケットリレー式にデータを送りあう初期のネットワークしかなかったが、インターネットによって世界中が即時につながり、地理的に分散した開発者たちの協働が可能になった。これはフリーソフトウェア財団が設立された頃には不可能なことである。

Linux の開発を原型として、オープンソース・ソフトウェアの開発は基本的にインターネット上に作業中のソースコード管理場所を持ち、メーリングリストを使って相互にコミュニケー

6) GNU プロジェクトがオペレーティングシステムを支える周辺のソフトウェアから足場固めをするもののオペレーティングシステム本体の着手に躊躇している間に Linux が現れた。

Linux はオペレーティングシステムの中核のみを開発し、周辺は GNU プロジェクトの成果を活用している。GNU プロジェクトの成果なしで Linux はここまで普及できなかったことは間違いない。フリーソフトウェア財団では、組み合わせて利用しているのだから単に Linux というのではなく、GNU/Linux というように表記することを求めている。

7) Torvalds, L. and Diamond, D., *Just for Fun: The Story of an Accidental Revolutionary*, 2001 (風見 潤 訳『それがぼくには楽しかったから』小学館プロダクション, 2001年)

8) オープンソースプロジェクトへの参加動機を調査したものとしては、OSDN (Open Source Technology Group) と Boston Consulting Group が 2002 年に報告した Hacker Survey (<http://www.ostg.com/bcg/>) や、International Institute of Infonomics が 2002 年に実施した Free/Libre and Open Source Software: Survey and Study (<http://www.infonomics.nl/FLOSS/report/>) などがある。

Torvalds の「それがぼくには楽しかったから」のように、知的好奇心を満足させるものもいれば、ソフトウェア開発能力の向上、自分の評判を高める、自分が作りたいソフトウェアを作る、そして、コミュニティでの評価を高めたい、などが参加動機であると答えるものもある。評判を高めるためには、参加するプロジェクトが著名である必要があるが、実際のところ進行中のオープンソース・ソフトウェア開発プロジェクトは、数万ある。ほとんどは、無名のプロジェクトである。したがって、多くのプロジェクト参加者たちは、自らの使いたいソフトウェアが世の中にはないのだが、自分一人で作るの大変だ、というような考えで参加していることになる。つまり、このような動機付けを中心に、知的好奇心の追求をまで含め、参加者は社会のためにソフトウェアを開発しているのではなく、参加者個人の自己利益を追求するためにプロジェクトに参加している。

ションをとることで行われている。その開発組織は、それぞれの動機付けがさまざまであったとしても基本的には自発的に開発プロジェクトに参加した個人で構成され、そこには明示的な指揮命令の系統はない。このような特徴から、オープンソース・ソフトウェアの開発組織はコミュニティとして扱われることが多い。コミュニティの統制は、主にプロジェクト開始者が発揮するリーダーシップと相互信頼とによる。

明確な分担が行われない代わりに潤沢な開発要員のため、同一部分を複数で並行開発し、その成果の中からよいものを選択するという冗長性の高い開発が可能なことから、結果として高品質のソフトウェアを開発する仕組みになっている。

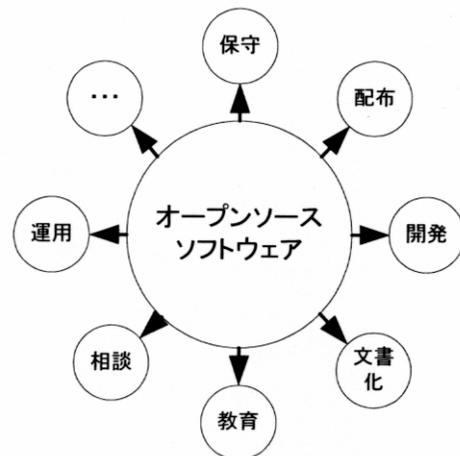
3. オープンソース・ソフトウェアとビジネスとの関わり

オープンソース・ソフトウェアとビジネスとの関わりを詳述する前に、これらの関係について歴史的な移り変わりを整理しておく。ここでは、「寄生モデル」、「共生モデル」、「一体化モデル」と名付けた 3 つのモデルを提案する。これらのモデルは、紹介する順で出現しているが、新しいモデルが古いモデルを駆逐して支配的になることはなく、共存している。すなわち、オープンソース・ソフトウェアとビジネスとの間の関係は、新たな関係モデルを生み出しながら多様化しているといえる。

(1) 寄生モデル

寄生モデルは、ソフトウェア開発機能を中心としたオープンソース・ソフトウェア開発コミュニティの機能を補完するかたちで、ビジネスは利益を得ている。従来からの一般的なソフトウェア製品についても、同じようなビジネスモデルが可能であるから、オープンソース・ソフトウェアを中心として伝統的なソフトウェア・ビジネスを行っているといつてよい。自由に配布されるオープンソース・ソフトウェアを巡るビジネスは、差別化が困難であり大きな利益を上げることは困難である。

周辺で活動するビジネスが、中心となるオープンソース・ソフトウェアの開発に貢献をしたり、影響を与えたりすることはほとんどないが、中心となるオープンソース・ソフトウェアに自力では接近しがたいユーザへの支援を提供し、ユーザ層を充実させるかたちでオープンソース・ソフトウェアの開発コミュニティに対して貢



図表1 寄生モデル

献する可能性がある。ユーザが増えれば、社会的な認知が高まり、コミュニティに参加して開発に貢献しようとする開発者の誘因が高まる。この貢献は、あくまで副次的なものであるので、ここでは寄生モデルと名付けた(図表1)。

(2) 共生モデル

オープンソース・ソフトウェアが普及し始めると、オープンソース・ソフトウェアを本質的にビジネスに取り込むことで、差別化を図り、戦略的優位性を獲得しようという動きが出てきた。

既存のソフトウェアをオープンソース化して競争力を付けようとした Netscape Communications の動きは、ようやく最近になってシェアを獲得し

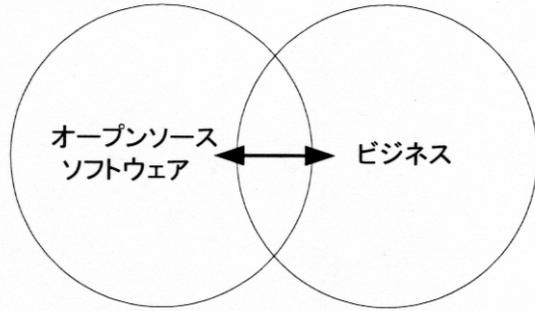
始めてきている。また、Linux オペレーティングシステムの開発に社員を業務として投入している IBM は、自社の全システムで Linux が稼働することによって、Linux をプラットフォームとしたビジネスでのリーダーとなることを目指している。日本において IBM は大型汎用計算機売上の 20~30% を Linux 搭載機が占めるようになり、その成果が現れてきている⁹⁾。

また、IBM は、Linux に注力することで、市場で支配的な Microsoft 製品の地位を相対的に低下させることも目指している。

(3) 一体化モデル

最後のモデルは、まだ表面化したばかりで事例も少ないが、今後の普及の可能性が期待できる。

多くのオープンソース・ソフトウェアは、オペレーティングシステムや汎用のユーティリティなど、ユーザ層の厚さが期待できる対象で開発されることが多かった。これは、「開発者=ユーザ」という構図で開発者を確保することに由来している。新しい形態では、ユーザ数が期待できないビジネス・



図表2 共生モデル



図表3 一体化モデル

9) 2004年度のzSeries(汎用機シリーズ)の売上は、前年比2桁増。Linux搭載機は同40%増で、zSeries全体の売上増に貢献している。この傾向は2005年度も継続している。

(日経IT Pro <http://itpro.nikkeibp.co.jp/free/ITPro/NEWS/20050701/163816/>)

アプリケーションについてのオープンソース化を進めている。この場合、ソフトウェアを共有することで、保守、強化のコストを案分し、軽減することが目的である。ビジネス・アプリケーションでは、開発から導入までのコスト以上に、運用や保守にコストがかかることは常識となっている。本稿で取り上げるニュートキョーのセルベッサの場合では、当初、コスト低減を目的としてオープンソース化を行ったが、オープンソース化によるインタフェイスの共有化が、新たな企業間ネットワーク・システムを構築することにつながっている。

一企業の投資によって開発されるビジネス・アプリケーションをオープンソース化することには、種々の困難があるが、戦略的な差別化に関わらない業務については、オープンソース化によるコスト軽減は今後の企業情報戦略において一つのオプションとなることが期待できる。

4. オープンソース・ソフトウェアを巡る伝統的ビジネス（寄生モデル）

GNU GPL ライセンスで保護されたオープンソース・ソフトウェアを巡るビジネスは、ソフトウェアの周辺で利用者を支援する活動から始まった。オープンソース・ソフトウェアを利用したいが、自分だけでは困難な状況にいるものを有償のサービスで支援するビジネスである。具体的に主要なものには、ソフトウェアの配布、利用にあたってのコンサルティング、機能変更のためのソフトウェア開発、運用の支援などがある。

（1）ソフトウェアの配布

オープンソース・ソフトウェアの多くは、インターネット上に公開され、誰でも自由に入手できるようになっているが、ソフトウェアによっては開発中の版などが混在して公開され、また、ソフトウェアによっては、利用するために様々なソフトウェアをそれぞれの Web サイトから入手して組み合わせなければならないものもある。ソフトウェアをインターネット上から入手できても、それをコンピュータに設定するために知識が要求されるものもある。

そのようなソフトウェアをまとめて CDROM などに記録し、設定のための機能を追加して販売しているものがある。たとえば、Linux オペレーティングシステムに周辺のソフトウェア・ツール群をまとめ、適当な期間の問い合わせ対応などのサポートと共に、日本語化機能の追加や設定作業を容易にするための機能を追加して販売しているパッケージは各種見ることができる。

これらのほかに、様々なオープンソース・ソフトウェアを収集し、CDROM などに記録して販売しているものもある。しかし、Linux であれ、ほかのオープンソース・ソフトウェアであれ、インターネットを通じて入手できるものを基本としているのであるから、差別化は困難であり、競合同士での価格競争が発生し、高い収益を得るビジネスとはなりにくい。

(2) コンサルテーション

オープンソース・ソフトウェアは一般にマニュアルなどの文書類が十分に整理されていないことが多い¹⁰⁾。このために、ビジネスで活用するためには、ノウハウの提供をうけなければならない場合がある。

まず、問題解決のためにどのようなソフトウェアの活用が有効かを検討しなければならないが、オープンソース・ソフトウェアまで範囲を広げると、すべてが実用に耐えるものではないにしても、約十万種のソフトウェアが存在している¹¹⁾。適切なソフトウェアを探し出して、評価する時点で知識の提供がビジネスとなる。

さらに、マニュアル化が十分ではないソフトウェアを使う場合には、ソースコードを調査して機能を確認しなければならない場合がある。これを代行するビジネスも可能である。ただし、この場合、ソースコードを調査した上で要求にあわせて改変を行うことが想定され、ソースコードに手を入れるビジネスは、次の受託開発に分類できるだろう。

(3) オープンソース・ソフトウェアをもとにした受託開発

オープンソース・ソフトウェアはGNU GPLによって、ソースコードが公開されている。公開されているソースコードは内容を読んで機能を調べるだけでなく、機能が不十分であれば独自に改変して機能追加をした上で利用することも許されている。

ソースコードの改変を自らでできない場合には、改変について対価を払って委託することになる。GNU GPLでは、このような形での受託開発ビジネスを認めている。ソフトウェア開発を発注する側からすれば、オープンソース・ソフトウェアを出発点とすれば、何も無いところからの開発に比べて、コストの削減と品質の向上が期待できる。

オープンソース・ソフトウェアの改変にあたってはライセンス条項に注意が必要になる。たとえば、GNU GPLでは、GPLで保護されるソースコードを改変した場合、その再配布にはGPLを適用することを義務づけている。すなわち、経費をかけて改変をしたとしても、改変後のソースコードをソフトウェアにあわせて配布しなければならない。また、別途独自に開発していたソフトウェアをオープンソース・ソフトウェアの改変に組み込んだ場合、独自開発して

10) Linuxのような大きなプロジェクトでは、ソフトウェアの開発に並行してマニュアルなどの文書を作成するプロジェクトが存在することもあるが、多くのオープンソース・ソフトウェアでは、マニュアルをするよりもソフトウェアの開発に注力している。また、ソフトウェアの更新が頻繁なので、マニュアルが追いつかないこともある。

11) 2005年7月10日現在で、オープンソース・ソフトウェア情報を整理しているサイト <http://sourceforge.net> に登録されているプロジェクトが102,818件、日本のローカルなオープンソース・ソフトウェア情報を整理しているサイト <http://sourceforge.jp> に登録されているプロジェクトが1,588件ある。このほかに、独自のサイトでソースコード管理を行っているプロジェクトも多数ある。

いた部分のソースコードまで公開しなければならなくなる。これを避けるためには、ソフトウェアの境界を明確にして開発しなければならない。

オープンソース・ソフトウェアをもとに独自開発あるいは受託開発する場合には、その際の投資を保護する方法を考えなければならないかもしれない。Stallman は GNU GPL のなかに、自由なソフトウェアが拡散する仕組みを作っていたのである。

(4) 運用支援

Linux のようなオペレーティングシステムを利用してサーバを運用する場合には、ハードウェアを用意して運用することが一般的であるが、サーバを運営・管理するためには、要員の配置・教育が必要になる。そこで、Linux に限らず Web ホスティングや電子メールサーバなどの機能をインターネット経由で提供することがビジネスになる。

提供する機能が、どのようなものであってもサービスのコストを下げ収益を向上していくためには、サービス提供のインフラ経費を低くすることが重要である。このため、インターネット経由でのサーバ機能提供サービスでは、オープンソース・ソフトウェアとして公開されているソフトウェア群を組み合わせていることが多い。

(5) 保守

オープンソース・ソフトウェアの開発は、初期のソースコードに改変や新たなソースコードの追加を繰り返して、機能強化していくことで行われていく。初期ソースコードの不備を修正しながら、機能追加と品質向上を行っているという意味では通常のソフトウェア・ビジネスにおける保守に近いという見方もある¹²⁾。しかし、開発者の立場からすると、この作業は開発者の問題意識によって動機づけられているので、あくまで開発であって、利用者の都合に合わせた保守作業ではなく、オープンソース・ソフトウェアの開発コミュニティでは保守に対する開発者の動機付けが困難である¹³⁾。

そこで、業務システムにオープンソース・ソフトウェアを利用しようとする場合には、自らでソースコードのレベルでの修正ができなければ適切な保守体制を委託しなければならない。導入コストが安いからといって、万一の場合に備える体制が整わなければ本格的に業務システムに利用することは困難である。

12) 佐藤一郎「オープンソースの理想と現実」(2003年6月4日)

<http://www.rieti.go.jp/it/column/column030604.html> 2005.03.07 アクセス

13) 八田真行「オープンソースの現実、と若干の理想(上)」(2003年6月28日)

<http://japan.linux.com/opensource/03/06/28/1235243.shtml> および「オープンソースの現実、と若干の理想(下)」(2003年8月1日) <http://japan.linux.com/opensource/03/07/31/190257.shtml> 2005.03.07 アクセス

Linux の配布サービスの中には、配布メディアの中に初期の保守サービスを含め、以降は別途有償で保守サービスを提供しているものもある。

以上のようなビジネスは、オープンソース・ソフトウェアの周辺に従来からのソフトウェア関連ビジネスを展開しているものと見ることができる。いずれにしても、誰でもが自由に入手できるソフトウェアをその中心にしているのであるから、参入障壁は比較的 low、特別に差別化できる能力を持っていなければ、高い収益を期待することは難しい。

オープンソース・ソフトウェアの開発が広く展開することによって、ビジネス・コミュニティの中に、オープンソース・ソフトウェアの手法を取り入れる事例や、本業の差別化を図るためにオープンソース・コミュニティとの関係強化を進める事例が出現している。

5. オープンソース化される商用ソフトウェア（1）…（ビジネスとの競合）

情報産業における競争の主体がソフトウェアに移ってくると、競争の手段としてオープンソース・ソフトウェアの仕組みを利用する事例が出てくる。その代表例は、Netscape Communications による Netscape Navigator のソースコード公開である。

Microsoft の Internet Explorer という競争相手の出現によって、劣勢となる Netscape Communications であるが、ブラウザソフトのソースコードを公開することで、ソフトウェアは生き残り復活の兆しさを見せ、Microsoft の戦略に影響を与えるまでになっている。

（1）Netscape Navigator の歩み

Netscape Navigator は、WWW サービスのページを表示するためのブラウザである。インターネットの普及を支援したいイリノイ大学の NCSA¹⁴⁾ が開発し、1993 年に公開した Mosaic というブラウザが原型となっている。

Silicon Graphics の創業者である Jim Clark が出資し、Mosaic の中心的開発者であった Mark Andreessen を迎えて、1994 年に Netscape Communications を設立した¹⁵⁾。1995 年 6 月に正式版が公開された Netscape Navigator は無償で配布され、インターネットの普及にあわせて世界中で、80%以上の利用シェアを持ち、支配的なブラウザとして広く利用されるようになった。

その後、有償化と無償化を経て、Netscape Communicator としてメール・クライアントや

14) National Center for Supercomputing Applications

15) 当初は、Mosaic Communications であったが、イリノイ大学が Mosaic に対して商標権を主張したため、Netscape Communications に社名変更した。

ニュースリーダー機能を追加し、多機能化して現在に至っている。

（2）Microsoft Internet Explorer との競争

これに対し、Microsoft は 1995 年に独自のブラウザとして Internet Explorer を開発し、無償で配布を始めた。Netscape Navigator とほぼ同等の機能を持っていたが、のちに Windows オペレーティングシステムにバンドルして配布を始めると、一気に利用シェアを勝ち取っていった。

有償化した Netscape Navigator の販売ビジネスを考えていた Netscape Communications にとって、この状況の変化は再びソフトウェアの無償配布に方向転換を強いただけでなく、1998 年には AOL (America Online) に買収されてしまった。同じ 1998 年には、Microsoft に対して、オペレーティングシステムにバンドルしたブラウザの配布について独占禁止法違反を問う訴訟も起きている。

（3）オープンソース化

Microsoft との競争に対抗するために、Netscape Communications は、Netscape Navigator のソースコードを 1998 年に公開した¹⁶⁾。これによって、力を増してきていたオープンソース活動の力を借りて、ブラウザソフトの機能強化を図ろうとしたのである。ソフトウェアのオープンソース化は、また、ソフトウェアに対する安心を生むことにもつながる¹⁷⁾。オープンソース化された Netscape Navigator は Mozilla と名付けられ、Netscape Communications とは独立に、Mozilla.org (The Mozilla Organization) という組織が作られ開発活動の運営の中心となってきた。

オープンソース化の初期には、公開されたソースをもとに多数のバージョンが発生し、収拾がつかなくなることが危惧されたが、Mozilla.org の活動によって、最低限の調整がなされ細々とではあるが、オープンソース化された Mozilla ブラウザの開発と公開が継続していった。

状況が変化するのは、2003 年に AOL が出資して、Mozilla.org を Mozilla 財団 (Mozilla Foundation) に改組してからである。

16) Netscape Communications が Netscape Navigator のソースコードを公開したのは、AOL に買収される前であった。

17) ソースコードを公開していないソフトウェアでは、ソフトウェアの中で行われていることに対して、ソフトウェアを提供する企業を無条件で全面的に信用しなければならないが、ソースコードが公開されていれば、自らその内容を確認できる。自らできないにしても、公開されたソースコードを誰もが見られるという状況から、明らかな不正コードは作り込まれないことが期待される。

(4) 現在の Mozilla

資金を得た Mozilla 財団は, Mozilla の開発を進め, 成果としてブラウザとしての Firefox やメール・クライアントの Thunderbirdなどを公開している。

Internet Explorer が繰り返し明らかになるセキュリティホールに, 場当たりに修正を加えているのに対して, Firefox は最新のセキュリティ技術を持ち, オープンソース・ソフトウェアの強みである頻繁な更新によって, 安全性が高い。また, Internet Explorer は2001年に公開された6.0を原型として修正を繰り返しているのに対して, Firefox は最新のブラウザ技術を導入し, 使いやすさの面でも優位に立っている。

このため, 2004年後半のブラウザの利用シェアは大きく動いた(図表4)。2004年後半にFirefoxを含むMozilla/Netscape系のブラウザ利用シェアは倍近くに急進している。数字はまだ小さいものの, Internet Explorer に機能的な変化がなければ, さらに数字をのぼしていくことが予想される。これに対抗するため, Microsoft は当初2006年に投入予定であったInternet Explorerの次期バージョンを2005年内に先行して投入し, 安全性が強化されたブラウザを志向する顧客に対応することを発表している。

オープンソース化されたブラウザと, オペレーティングシステムにバンドルされたブラウザ, この競争はまだ始まったばかりであるが, オープンソース・ソフトウェアの強みである頻繁な更新によって, 常に安心して高機能のブラウザが提供され続けられれば, インターネットの常時接続が普及し, 簡便にインターネット上からソフトウェアを入手できる状況では, 競争関係が大きく変化していくことも考えられる。

単位 (%)

	2004年 6月	2004年 10月	2004年 12月
Internet Explorer	95.48	93.21	91.8
Firefox		2.66	4.06
Firefox 以外の Netscape/ Mozilla	3.58	3.05	2.83
その他	0.95	1.03	1.25

図表4 米国内のブラウザソフト利用シェア¹⁸⁾

18) 朝日新聞 2005年2月17日より。2004年6月時点では, Firefox とそれ以外の Netscape/Mozilla ブラウザをあわせて集計している。

6. 戦略的地位構築手段としてのオープンソース・ソフトウェア(共生モデル)

ブラウザにおける競争は、ソフトウェア・ビジネスに関する競争であった。しかし、情報産業での競争の主戦場は、ソリューション・ビジネスに移行しようとしている。その中心は、主力ビジネスをハードウェアからソリューションへと劇的な転換を行った、IBM といつてよいだろう¹⁹⁾。IBM は、また、オープンソース・ソフトウェアとの関わりによって、自らの戦略的地位の向上を目指している。

(1) Linux との関わり

IBM が Linux の開発コミュニティに人員や資金の提供を始めたというニュースは、世間を騒がせるものであったが、状況が明らかになるにしたがって、ソリューション・ビジネスを追求する姿勢が明らかになっている。

IBM のハードウェアが市場で支配的であったときには、IBM が自ら技術標準を提案することが可能であったが、1980年代以降、システムがオープン化²⁰⁾していくと、標準は市場によって作られていくようになる。IBM は、2001年にはLinuxビジネスに10億ドルを投資し、以降その額は増える一方である。投資は、Linux開発コミュニティに関わる開発者への資金提供を始め、社内のLinux関連ビジネス組織の整備にあてられている。特に、Linuxカーネル²¹⁾の開発には力を入れている。そして、IBMの提供する全プラットフォームでLinuxが稼働する状況を作り出そうとしている。

LinuxはIntel系のCPUを持つコンピュータを中心に開発が進められてきているが、本格的にビジネス用途で活用するためには、さらに高速で信頼性の高いコンピュータでの利用が求められる。IBMは、開発コミュニティに深く関わることで、自社の持つメインフレームを含む全ハードウェアでLinuxが稼働する環境を手にした。単一のオペレーティングシステムで全ハードウェアを統一するということは、IBM自身の手によってもこれまでなされたことがなかった。

結果として、IBMは最上位の大型コンピュータまでシームレスにLinux関連サービスが提供できるビジネスを展開できるようになった。

この経過で、もちろんLinuxに関するノウハウを蓄積することができた。シームレスなプラッ

19) パーソナルコンピュータ・ビジネスの売却は、コモディティ化したハードウェア・ビジネスからの決別を象徴する出来事といってもよいだろう。

20) システムがオープン化されるとは、システムの接続仕様が公開され、さまざまなシステムを相互に接続して利用できることである。そのようにして構成されたシステムをオープンシステムという。

21) カーネルとはオペレーティングシステムの中核部分である。一般にオペレーティングシステムは、カーネルとその周囲を構成するソフトウェアのセットであるが、Linuxの場合、周囲はほとんどGNUプロジェクトの成果であるから、カーネルはLinuxそのものといってもよい。

トフォームだけでなく、ソリューション・ビジネスに重要なノウハウの蓄積も並行して行われている²²⁾。

(2) 相対的優位性の獲得

集中的な投資によってLinuxの地位を向上することは、その競争相手となっているMicrosoftのオペレーティングシステムの競争的地位の低下につながる。

情報産業においては、IntelとMicrosoftがプラットフォームを支配する状況が続いている。コンピュータの頭脳といわれるCPUでは、互換メーカーがあるもののIntelがほぼ独占している。販売されているオペレーティングシステムについては、競合メーカーはなくMicrosoftが独占しているといってもよいだろう。

IBMのようにもともとハードウェア、オペレーティングシステム、アプリケーションなどのすべてを提供していたビジネスにとっては、このような状況は好ましくない。収益力の低下によって、ビジネスの構造をソリューション中心に変えなければならなくなったほどである。ソリューション・ビジネスに収益の中心を移したとはいえ、そのプラットフォームとして、Microsoftの提供するブラックボックス化されたオペレーティングシステムを利用することは、戦略的に好ましいことではない。

そこで、オペレーティングシステムの利用シェアとして、Linuxの地位を向上させることでIBMは結果として、Microsoftの市場での地位を低下させることが可能になる²³⁾。IBMは、Linuxビジネスに投資し、開発コミュニティに深く関わることによって、市場の中でのLinuxの地位を高めると共に、LinuxビジネスにおけるIBMの地位を高めることをねらっている。一方で、MicrosoftはこのようなLinuxをはじめとしたオープンソース・ソフトウェアの動きが無視できなくなっている。

22) Linuxの開発コミュニティに深く関わることによって、想定外の事態も起きている。SCOによる訴訟はその一つである。LinuxはUnixオペレーティングシステムを原型としているが、独立に開発を進めることによってライセンスの問題を避けている。IBMはUnixライセンスに基づいてAIXというオペレーティングシステムを開発し販売しているが、Linuxの開発にあたって、Unixのソースコードが転用されているとして、Unixのライセンス権者であるSCOが2003年に訴訟を起こしている。

オープンソース・ソフトウェアの開発コミュニティとの関わりを強化しようとしているIBMは、オープンソース・ソフトウェアの開発でソフトウェアに関する自社の特許事項などが制約とならないよう、2005年に自社の持つ特許500件をオープンソース・ソフトウェアの開発に利用する場合には無償提供することを表明している。

23) IBMだけでなく、オープンソース・ソフトウェアの開発に参加する開発者の中には、独占的にブラックボックス化されたソフトウェアを供給するMicrosoftへの反発が動機付けとなっているものが少なくない。

7. オープンソース化される商用ソフトウェア(2)...(一体化モデル)

ブラウザのように一般的に利用されるソフトウェアだけでなく、アプリケーションといわれる業務システムの中にも、オープンソース・ソフトウェア化されるものが出てきている。アプリケーション・ソフトウェアの多くは特定の企業が開発し、その企業が使用するものであるが、これをオープンソース化することで機能強化や、保守のコストを削減することが第一の目的となる。ソースコードを公開することはノウハウの流出につながる場合もあるので、ビジネスの中核に関わるアプリケーションをオープンソース化することはない。コスト削減や効率化を目指したアプリケーションの運用をさらにコスト削減や効率化を図るためにオープンソース化される。

オープンソース化されたアプリケーションは、当初の目的を超えて企業間の新たな関係形成につながるものも出てきている。その事例として、ニユートーキョーが開発して公開したセルベッサを見る。

(1) ニユートーキョーのセルベッサ

外食産業のニユートーキョーでは、各店舗からの食材発注を管理するシステムとして、システム開発を専門とするテナート二に委託してセルベッサを開発し、1999年7月に稼働させている。開発の途上でニユートーキョーとテナート二はオープンソース・ソフトウェアとして公開することを検討し始めたが、実際に公開されたのは稼働から4ヶ月間たち、実用に十分耐えることを確認した1999年11月である。オープンソース・ソフトウェアとして公開した動機は、改善や保守のコストを分担することによる軽減、ソースコードを通じたアプリケーションのチェック、委託先への依存関係の軽減の三つがあった²⁴⁾。

セルベッサは、ニユートーキョーの各店舗からの食材発注をまとめ、食品卸の三友小網に送り、店舗ごとの食材をまとめ一括発注できるようにしている。また、オープンソース化にともなって、サーバを設置して独自運用できない利用者のため、三友小網の親会社である三井物産がASPサービスの提供を開始した。

オープンソース化されたセルベッサは、定食屋大手チェーンの大戸屋をはじめ複数の外食チェーンで導入されている。三友小網を主な食材調達先として運用を開始しているが、利用企業の中には、三友小網との取引のないものもある。セルベッサからの受注可能な食品卸は、2005年3月時点で90社を超えている。

すべての外食産業企業と食品卸同士が取引関係を持つわけではないが、ここにセルベッサと

24) 竹田, 米山(2002) pp.152-153.

いう共通システムを仲立ちとした一つの企業ネットワークが構成されたと見ることができる。セルベッサが、このように展開していくことができたのは、本業の中核となるノウハウとはあまり関係のない分野で、多くの同業者にとって共通に必要な機能を提供しているからである。導入する外食産業企業と食品卸が増加していけば、このネットワークはさらに拡大、強化されていくことが期待される。

(2) オープンソース・ソフトウェアによる企業間ネットワークの形成

情報ネットワークを活用して企業間の取引ネットワークを形成しようとするとき、これまで、そのネットワークの中心となる企業がデータ通信のプロトコル²⁵⁾を提示し各企業にシステム開発をゆだねるか、受発注に必要なシステムを配布することが多かった。

前者の場合、システム開発能力がなければそのネットワークに参加することができないし、後者の場合、通常は配布された受発注システムがブラックボックス化されているので、自社の在庫管理システムなどとの連携は人手で行わなければならない。したがって、中心となる企業が市場で支配的な力を持ち、その企業との取引が戦略的に重要である場合に周辺企業がコスト負担をして企業間ネットワークが形成されていた。

セルベッサの事例は、新たな取引ネットワーク形成の可能性を開いている。オープンソース・ソフトウェアではあるが、稼働するソフトウェアとして提供されているので、上記の后者のようにブラックボックス化したソフトウェアとして活用して容易にネットワークに参加することが可能である。また、上記の前者のように自社の在庫管理システムなどとの連携を考える場合には、オープンソース・ソフトウェアの強みを生かして、ソースコードを改変してインタフェイスすることも可能になる。独自に開発するのに比べて大幅に少ないコストですむ。すなわち、参入障壁の低い企業間ネットワーク構成の手段として、オープンソース・ソフトウェアが活用できることを示している。

8. まとめ

本稿では、オープンソース・ソフトウェアは自発的組織によって開発されてきていることに関する背景などを概説した上で、ビジネスとの関係について、3つのモデルを提案し、事例をもとに整理を行った。

この関係の変化には、情報産業の主戦場がハードウェアから、ソフトウェアへ、そしてさらにソリューションへと移行している現実、さらにその移行を加速させようとする企業の力が働いていた。

25) データのやりとり、データの構成などを定めた規約のこと。

まず、オープンソース・ソフトウェアと一般の市販されているソフトウェアとが独立した存在であったときには、オープンソース・ソフトウェアが一般の市販ソフトウェアに対して欠けている部分を補うビジネス(寄生モデル)が、特に利用者の多いオープンソース・ソフトウェアの周辺に生まれた。これは、現在も継続している。ソフトウェアの配布、コンサルテーション、受託開発、運用、保守などである。これらのビジネスは、いずれも誰でもが自由に入手できるオープンソース・ソフトウェアそのものに依存しているために、特に大きな差別化要因を持っていない限り、高収益を生み出すことは困難である。

第二に、オープンソース・ソフトウェアと市販ソフトウェアとの間の競争は、Web ブラウザに見られる。Microsoft はオペレーティングシステムへのバンドルを通じて、手軽に入手でき利用できるという利点によって、短期間のうちに利用シェアを勝ち取ったが、オープンソース化された Netscape Navigator のバージョンである Firefox が登場すると、Internet Explorer の安全性や機能に不満を持つ利用者を短期間に集め始めている。この競争の結果、Microsoft は自らの製品計画を変更しなければならなくなっている。ブラウザの競争はまだ始まったばかりであり、今後も激しい競争が展開されると予想される。

第三に、オープンソース・ソフトウェアの開発コミュニティと企業との共同によって新たな戦略的価値を作り出すことが可能である。IBM は、Linux 開発コミュニティに集中的に多額の投資をすることによって、Linux 市場における IBM の競争的地位を向上すると同時に、資金を得た Linux 開発コミュニティは開発を充実し、オペレーティングシステム市場における戦略的地位を向上する(共生モデル)。Linux の地位向上は一方で Microsoft の相対的な地位低下を意味し、これは IBM の望むところである。オープンソース・ソフトウェア開発コミュニティと企業とが、それぞれの目的と共通の目的を整合させて、協働する可能性を示している。

最後に、オープンソース化されたアプリケーションが、企業間ネットワークを構成する新たな枠組みを提供する可能性がある。ニユートーキョーは、独自の食材発注システムセルベッサを公開することで、機能強化、保守の経費分散を図った(一体化モデル)が、結果として競合ともいえる外食産業他社や多数の食材卸を巻き込んだ緩やかな企業間取引ネットワークの形成を可能にした。このネットワークへの参加は、セルベッサをダウンロードして利用するだけであるから参入障壁は非常に低い。

オープンソース・ソフトウェアの前身ともいえるフリーソフトウェアの誕生から、まだ、20年あまりの歳月しかたっていない。この間、ビジネスとの関わりは、オープンソース・ソフトウェアの周辺の小規模のベンチャーを除くと、一部特定企業間の競争関係や、より複雑な協働関係の構成が見られるようになったが、それは、この10年以内のことである。

実用的に使用できるオープンソース・ソフトウェアの多様性が高まり、社会への浸透を深め

ていくにしたがって、本稿で取り上げた関係よりも新しいさまざまな関係が形成されていくことが期待される。これらの可能性を検討すると共に、新たな関係を整理していくことは今後の課題となるだろう。

参考文献

- 朝日新聞「閲覧ソフト戦線 異変 無料ファイアーフォックス台頭」(2005年2月17日)
- 八田真行「オープンソースの現実, と若干の理想(上)」(2003年6月28日)
<http://japan.linux.com/opensource/03/06/28/1235243.shtml> 2005.03.07 アクセス
- 八田真行「オープンソースの現実, と若干の理想(下)」(2003年8月1日)
<http://japan.linux.com/opensource/03/07/31/190257.shtml> 2005.03.07 アクセス
- International Institute of Infonomics, *Free/Libre and Open Source Software: Survey and Study*,
<http://www.infonomics.nl/FLOSS/report/> 2005.03.07 アクセス
- 日経 IT Pro「IBM 汎用機の2~3割がLinux搭載機に、選ばれる理由は意外な“割り引き”効果」(2005年7月4日)
<http://itpro.nikkeibp.co.jp/free/ITPro/NEWS/20050701/163816/> 2005.07.10 アクセス
- ニュートキー「セルベッサ」<http://www.newtokyo.co.jp/cerveza/> 2005.03.13 アクセス
- OSTG, The Boston Consulting Group/OSTG Hacker Survey,
<http://www.ostg.com/bcg/> 2005.03.07 アクセス
- 佐藤一郎「オープンソースの理想と現実」(2003年6月4日)
<http://www.rieti.go.jp/it/column/column030604.html> 2005.03.07 アクセス
- 竹田陽子, 米山茂美「ビジネス・ケース セルベッサ ニュートキーの食材発注システムはなぜ公開されたのか」『一橋ビジネスレビュー』2002. Win. pp.146-165.
- Linus Torvalds and David Diamond, *Just for Fun: The Story of an Accidental Revolutionary*, Harpercollins, 2001(風見 潤 訳, 『それがぼくには楽しかったから』, 小学館プロダクション, 2001).