

Title	オープンソース方式におけるイノベーションプロセスの分析 - 大学発オープンソースソフトウェアの可能性 -
Author(s)	杉本, 宏史
Citation	
Issue Date	2002-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/348
Rights	
Description	Supervisor: 亀岡 秋男, 知識科学研究科, 修士

修 士 論 文

オープンソース方式におけるイノベーションプロセスの分析
—大学発オープンソースソフトウェアの可能性—

指導教官 亀岡秋男 教授

北陸先端科学技術大学院大学
知識科学研究科知識社会システム学専攻

050044 杉本 宏史

審査委員： 亀岡 秋男 教授（主査）
永田 晃也 助教授
梅本 勝博 助教授
遠山 亮子 助教授

2002 年 2 月

目次

第1章 研究目的・方法.....	1
1-1 研究の背景.....	1
1-1-1 "製品"としてのオープンソースソフトウェア.....	1
1-1-2 大学との関わり.....	2
1-2 研究の目的と方法.....	3
1-2-1 本研究の目的.....	3
1-2-2 研究方法と論文構成.....	3
第2章 オープンソースソフトウェアの概要.....	4
2-1 はじめに.....	4
2-2 オンラインソフトの種類.....	5
2-2-1 オンラインソフトとは.....	5
2-2-2 フリーソフト.....	6
2-2-3 フリーウェア.....	7
2-2-4 オープンソース.....	8
2-3 オープンソースの特徴.....	11
2-3-1 オープンソースの特徴とは.....	11
2-3-2 複製・配布の自由.....	11
2-3-3 ソースコード入手の自由.....	12
2-3-4 修正・改良の自由.....	13
2-4 OSS とフリーウェアの違い.....	13
2-5 オープンソースにおけるライセンス形態.....	14
2-5-1 オープンソースのライセンス形態とは.....	14
2-5-2 GPL.....	15
2-5-3 X,MIT,BSD.....	16
2-5-4 NPL.....	17
第3章 OSS 開発における原動力の分析.....	19
3-1 分析の手法.....	19
3-2 デジタル財.....	20
3-2-1 デジタル財とはなにか.....	20
3-2-2 デジタル財の公共財的側面.....	20
3-2-3 OSS との関係.....	21

3-3 ネットワーク	21
3-3-1 二つの「ネットワーク」	21
3-3-2 コンピュータとしてのネットワーク	22
3-3-3 コミュニケーションとしてのネットワーク	23
3-4 ボランティア	24
3-4-1 ボランティアとは	24
3-4-2 OSS コミュニティとボランティア	24
3-4-3 ボランティアコモンズ	24
3-4-4 企業戦略としてのボランティア	25
3-5 分析結果	26
第4章 大学と OSS に関する調査	28
4-1 調査方法	28
4-2 ディストリビューション調査	28
4-2-1 調査内容	28
4-2-2 調査対象	29
4-2-3 調査方法	30
4-2-4 調査結果	31
4-2-5 分析	34
4-3 事例研究	35
4-3-1 調査目的・方法	35
4-3-2 PostgreSQL とはなにか	35
4-3-3 PostgreSQL の歴史	37
4-3-4 PostgreSQL 発展の流れ	39
4-3-5 分析	39
第5章 結論	45
5-1 大学発 OSS におけるイノベーションプロセスの新規性	45
5-2 大学における OSS 活用に関する提言	45
5-2-1 デジタル財の性質を活用する	46
5-2-2 OSS が育つ環境の整備を	46
5-3 解明点	47
参考文献一覧	48

図表一覧

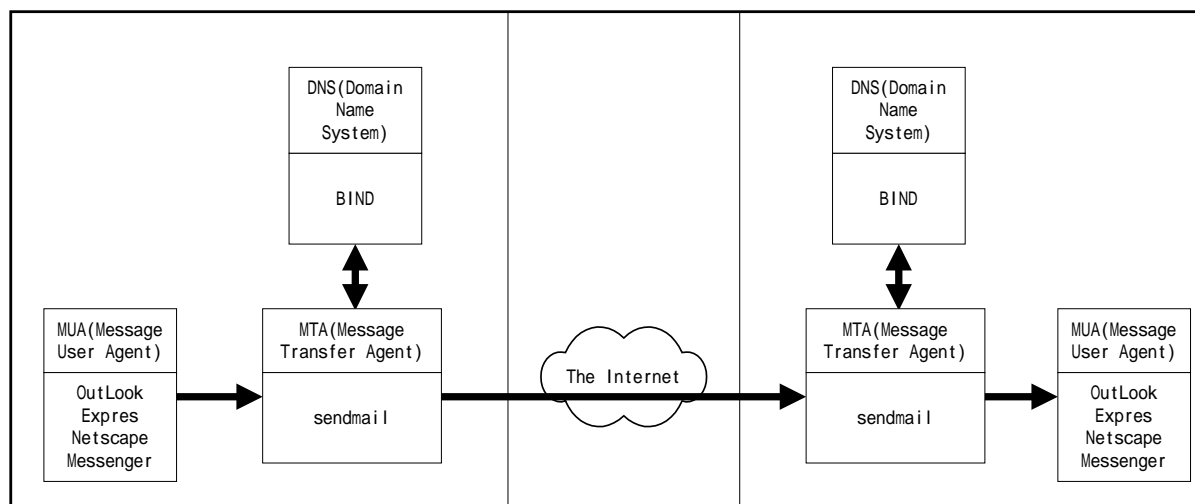
図 1-1 インターネット上における電子メールの流れ.....	1
図 2-1 流通ソフトウェア分類図.....	5
図 3-1 OSS における原動力.....	27
図 4-1 メールアドレス調査・トップドメイン内訳.....	31
図 4-2 メールアドレス調査 jp ドメイン内訳.....	33
図 4-3 PostgreSQL 系統図.....	39
図 4-4 Mosaic 系統図.....	40
図 4-5 従来の研究成果の移転プロセス.....	43
図 4-6 OSS による研究成果の移転プロセス.....	44
表 1-1 電子メールシステムで使用されるソフトウェア.....	1
表 2-1 OSS の一例.....	4
表 2-2 OSS ライセンス一覧表.....	15
表 4-1 主な Linux ディストリビューション一覧.....	29
表 4-1 図 4-1 におけるドメイン名称.....	31
表 4-2 図 4-2 におけるドメイン名称.....	33
表 4-3 OSS 開発主体.....	35
表 4-4 大学発 OSS 一覧.....	40

第1章 研究目的・方法

1-1 研究の背景

1-1-1 ”製品”としてのオープンソースソフトウェア

現在、インターネットの利用で最も広く使われているシステムは、電子メールとWWW(World Wide Web)といっても良いだろう。インターネット上の電子メールは、複数のシステムを経由して相手に送られている。一般に使用されているシステム構成は次のようになっている。



[図 1-1 インターネット上における電子メールの流れ]

ソフトウェア名	内容
netscape Messenger	ネットスケープ社のメーラー (MUA)
Outlook Express	マイクロソフト社のメーラー (MUA)
sendmail	広く普及しているメール転送エージェント (MTA)
bind	DNS (Domain Name System) サーバー
apache	インターネット上で最も一般的に使用されている Web サーバー
mozilla	Web ブラウザ

[表 1-1 電子メールシステムで使用されるソフトウェア]

このようにインターネット上で電子メールを相手に送るために、複数のシステムが連携して稼働している。ここで注目すべき点は、上記に挙げたソフトウェアは全て無償で利用できる点である。この点はWWWも同様である。WebサーバであるApacheや、WebブラウザであるマイクロソフトのInternet Explorer(IE)やネットスケープ

コミュニケーションズの **Netscape Navigator** なども無償で利用することができる。

このように無償のソフトウェアはインターネットの世界では広く使われているが、これには大きく分けて次の2つの種類に分類することができる。

ひとつは **IE** などインターネット上で流通する多くの商用ソフトのように、無償提供されていても、利用に制限がつけられていたり勝手に改良したりすることはできないソフトウェアである。これらはフリー（無償）で使えるが、フリー（自由）に使えないソフトウェアである。

もう一つは、**Linux** や **Apache** などのように無償で手にいれることができ、利用や改良が自由にできるソフトウェアである。これらはフリー（無償）でありかつ、フリー（自由）に利用できるソフトウェアである。この特徴は、企業が製品として開発したものではなく、個人やオープンソース・コミュニティと呼ばれる有志のグループが自発的に開発しているソフトウェアという点である。このような形態のソフトウェアはオープンソースソフトウェア（**OSS**）と呼ばれている。

OSS はインターネットのあらゆるところで活用されている。インターネットはすでに社会にとって不可欠なものといつてよいだろう。そして我々は **OSS** によって構築されたインターネットをすでに活用しているのである。

OSS について語るとき、コミュニティに目が奪われがちである。**OSS** がオープンソース・コミュニティによって開発され、その存在が不可欠なことは確かである。実際に **OSS** に関する研究の多くはコミュニティに重点がおかれているか(レイモンド [1997])(木村[2000])、ネットワーク上のビジネスモデルの視点から活用方法が検討されているもの(根来[1999])(國領[1999])が多い。

しかし **OSS** をソフトウェアという”製品”のひとつでもある。インターネットの多くが **OSS** によって構築されている現状を考えれば、**OSS** を”製品”としてどのようなプロセスで開発され、発展していくかについて注目する必要があるのではないだろうか。

1-1-2 大学との関わり

ソフトウェアは製品であると同時に、ソースコードという知識の集積と考えることができる。他の物理的な製品、たとえば電機製品ならば設計図を元に材料や部品を集めて組み立てることによって製品を製造する。しかし、ソフトウェアはいわばソースコードという設計図がそのまま製品になる。このように書かれた情報がそのまま製品になるのはソフトウェアの特質の一つである。プログラム言語で記述されたものであれば複写すること可能である。さらにソースコードに書き足すことによって機能の追

加も可能である。つまりソフトウェアは、コンピュータを動かすための知識をソースコードという形で記述することによって製品となるといえる。これらの視点から考察すると、**OSS** は多くの人の創造と知識を蓄積とした結果といえる。

本研究ではこのような **OSS** を生み出すきっかけとなるトリガーに注目する。開発したソフトウェアを **OSS** として公開する集団は大きく分けて個人もしくはコミュニティ、大学、企業の3つである。そのなかで注目しているのは大学の役割である。知識や技術の創造や蓄積という特徴は大学の役割と類似性がある。大学の研究成果と **OSS** の役割を分析することで、大学が **OSS** を活用する知見を得ることができるのではないだろうか。

1-2 研究の目的と方法

1-2-1 本研究の目的

本研究は **OSS** がどのように開発・発展していくプロセスに注目する。つまりソフトウェア製品の一つである **OSS** が、どのようなプロセスを経て発展するのかという側面からイノベーションの新規性を考察する。

本研究の目的は、**OSS** が持つイノベーションプロセスの新規性を明らかにすることである。そして大学が **OSS** を活用する方策を提案することである。

そのためには**(1)****OSS** が持つイノベーションプロセスの新規性を明らかにし、**(2)**これが大学研究成果の社会還元の手法になりうるかを調査する。

1-2-2 研究方法と論文構成

前述したとおり、**OSS** について包括的に述べた文献が乏しいのが現状である。そこで **OSS** についてははじめに把握する必要がある。そのためインターネット上の情報と文献を元に **OSS** の現状と概要を提示する（第2章）。次に **OSS** 発展の原動力について分析する。これは文献レビューを中心に行う（第3章）。さらに上記の成果を元に、大学と **OSS** との関わりについて調査を行う。調査方法は、**Linux** ディストリビューションの一つである **Vine Linux** の全ソースコードの分析と、**PostgreSQL** というオープンソースな **RDB** の事例研究を行う（第4章）。最後に結論として本研究で明らかになった点をまとめ、大学と **OSS** に関する方策を示す（第5章）。

第2章 オープンソースソフトウェアの概要

2-1 はじめに

現在オープンソースソフトウェア(以後 **OSS**)は、インターネットを中心に多くのシステムで活用されている。普及率の高いものには、**Linux** や **FreeBSD** などの **Unix** 系 **OS** や、**WEB** サーバ **Apache**、メール配信システムの **sendmail** などがある。このほかにも無数の **OSS** がインターネット上の様々なシステムに利用されている。

ソフトウェア名	内容	ライセンス
XFree86	Xワークステーション用基本フォント、プログラム、文書	XFree86
apache	インターネット上で最も一般的に使用されているWebサーバー	Apache Software License
bind	DNS (Domain Name System) サーバー	BSD-like
kernel	Linuxのカーネル (Linuxオペレーティングシステムの核)	GPL
mozilla	Webブラウザ	MPL
netscape-communicator	Netscape Communicator ツールセット	Proprietary
samba	Samba SMB サーバー	GNU GPL Version 2
sendmail	広く普及しているメール転送エージェント (MTA)	BSD
wu-ftp	ワシントン大学提供のFTPデーモン	BSD
postgresql	PostgreSQLのクライアントプログラムとライブラリ	BSD

[表 2-1 OSS の一例]

またインターネット上で開発された **OSS** は、インターネット以外のコンピュータシステムでも広く普及している。この中には、**OS** やアプリケーションソフトだけではなく、それらを開発するプログラミング言語も存在する。実際、**Perl**、**Ruby**、**Python** などのプログラム言語もオープンソースとして配布されており、多くの処理系に移植されている。

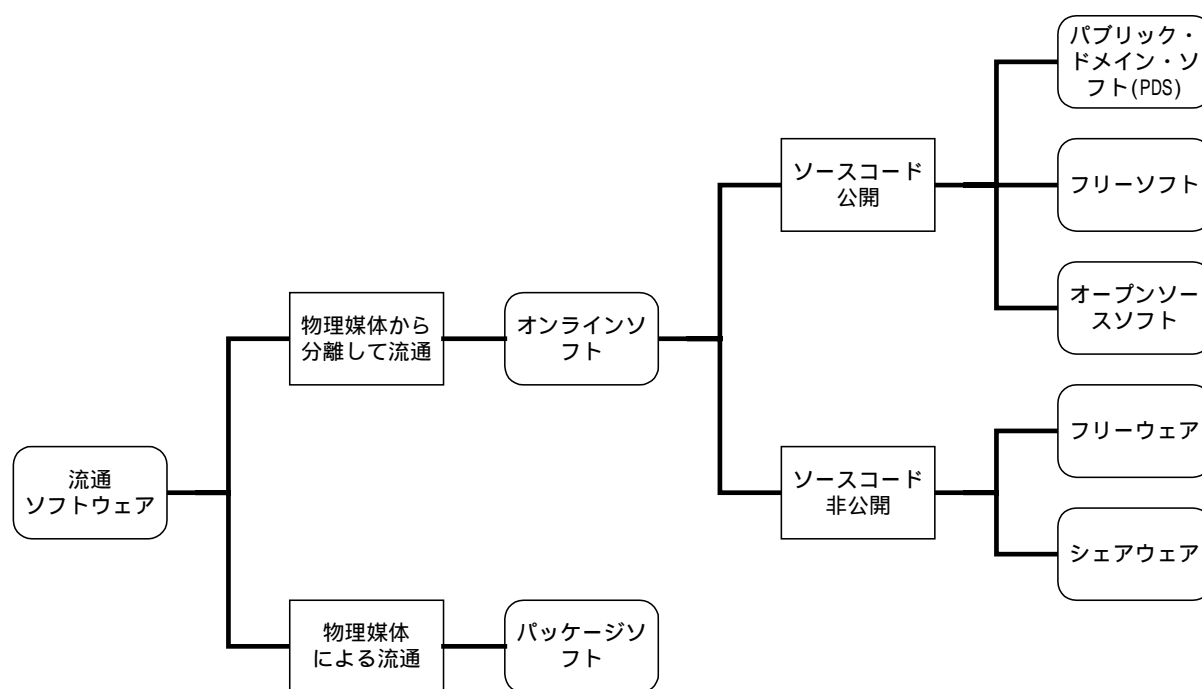
このように **OSS** は現実に情報化社会を構築する上で重要な役割を持つに至っている。しかし **OSS** は近年、急速に普及が進んだソフトウェア配布形式であり、**OSS** に関して述べられた文献は乏しいのが現状である。またネットワーク上に流通するソフトウェアの配布形式は **OSS** 以外にも存在するが、それらと **OSS** を区別する必要がある。

そのため **OSS** についての議論を後の章で行うためには、**OSS** の実体を把握する必要がある。そこで本章では **OSS** についての調査を行い、概要をまとめる。

2-2 オンラインソフトの種類

2-2-1 オンラインソフトとは

ネットワーク上に流通するソフトウェアの種類には、はっきりとした区分があるわけではない。これらはいずれも慣習的に誕生したもので、文脈によって意味合いが変わる場合がある。ここでは流通を意図して作成されたソフトウェアを次の図のように分類した。



[図 2-1 流通ソフトウェア分類図]

CD-ROMや**DVD-ROM**などの物理媒体に納めたソフトウェアを箱に入れて店頭で販売されているソフトウェアを、一般にパッケージソフトと呼ばれている。これに対してネットワーク上のコンピュータにあるソフトウェアを、自己のコンピュータにダウンロードして入手することのできるソフトをオンラインソフトと呼ばれている。

ここでは、オンラインソフトと呼ばれているソフトウェアをソースコード公開の有無で分類する。ソースコードを公開しているオンラインソフトとして、**PDS**・フリーソフト・**OSS**に分類できる。またソースコード非公開のオンラインソフトのうち、無

償で配布されているソフトをフリーウェア、有償で配布されているソフトをシェアウェアと呼ばれているケースが多い。

この節ではオンラインソフトのうち、フリーソフト、フリーウェア、**OSS** について説明する。

2-2-2 フリーソフト

コンピュータ黎明期において、ソフトウェアは独立した製品ではなく、ハードウェアの付属物として扱われていた傾向があった。また研究者やプログラマーは、互いにプログラムソースを譲り合ったりするのがごく当然に行われていた。それはソフトウェアを書くことがアカデミックな風潮があったためである。またプログラムが商品になるほど市場が成熟していなかったためでもある。

1970年代の風潮を後の**GNU**の創設者、リチャード・ストールマンは次のように述べている。

当時、「フリーソフトウェア」という表現はまだ存在していなかった
ので、我われは、自分たちのソフトウェアを「フリーソフトウェア」と
は呼んでいなかった。しかし、我われ他大学の人たちや会社の人たちが
プログラムを移植して使いたいと言ってきたときは、喜んでそうさせて
いた。目新しく面白そうなプログラムを使っている人がいれば、我われ
はその人からソースコードを自由に見せてもらうことができた。そう
いったソースコードの一部を自分たちで書き換えて使うこともできた。
そういうプログラムを好きなように解体して、別のプログラムの中で使
うこともできた。我われがソフトウェアをフリーソフトウェアとみなし
ていたことは事実である。 *[オープンソースソフトウェア p.104]*

しかし1980年代に入ると、ソフトウェアを製品として販売することが普通に行われるようになった。実際、**UNIX**のライセンスを持つ**ATT**が、これまで無償に近い形で提供していた**UNIX**のソースコードに高額のライセンス料を請求するようになった。また、マイクロソフトを初めとしたパソコン用のソフトウェア開発会社が登場したのもこのころである。

これらの企業が販売するソフトウェアは、プログラムの実行形式であるバイナリ形式で販売されていた。つまりソフトウェアの設計図ともいえるソースコードは、企業秘密でありユーザーが見ることはできなかった。

当時**MIT**人工知能研究所にいたリチャード・ストールマンは、このようなソフトウェアを独占的なソフトウェアと位置づけた。企業の独占的なソフトウェアは、ユー

ザー同士でプログラムの改善や機能追加などをする権利を奪い、利用者同士の協力や協調を奪うものと考えていた。

そこでストールマンは、独占的なソフトウェアを使わないで済むように全てのシステムが独占的ではないフリーなシステムの構築を開始した。それが **1983** 年に開始された **GNU** プロジェクトである。**GNU** プロジェクトとは "**GNU's Not UNIX**" の略であり、**UNIX** 互換 **OS** 開発のプロジェクト名である。そして **GNU** プロジェクトを推進するためにフリーソフトウェア財団(**Free Software Foundation,FSF**)を設立した。

初期の **GNU** プロジェクトはエディタ環境である **emacs**、**C** コンパイラである **GCC** など、主に **unix** 環境を中心に非常に高機能なソフトウェアを開発していた。そしてこれを磁気テープの配布や、学術用のネットワークであったインターネットを通じて配布し始めた。

GNU が主張するフリーとは、無償を意味するのではなく自由を意味すると繰り返し述べられている。そのため対価を受け取ることを否定してはいない。これはいわゆるタダのソフトではなく再配布・修正・研究が自由にできるソフトウェアを意味しているからである。

このような理念に基づいたソフトウェア配布形式がフリーソフトの特徴であり、後の多くのオンラインソフトに影響を与えている。

そしてこの理念を維持するために **GNU** プロジェクトでは、フリーソフトを **GPL** というライセンスを適用することで保護している。**GPL** について後に述べる。

2-2-3 フリーウェア

1980 年代に **GNU** の動きとは別に、パソコンの世界でも新しい形態のソフトウェアが誕生した。日本では総称してオンラインソフトと呼ばれている形式である。

パソコンの世界で使われるオンラインソフトには大きく **3** 種類ある。ソースコードも添付されていて著作権の放棄をうたった(日本では著作権は放棄できない)

PDS(Public Domain Software)。著作権は作者にあるが、利用・配布は自由であるフリーウェア。そして一定の試用後に少額の対価を求めるシェアウェアという形式である。また広義ではオンラインソフトの形態をフリーウェアと呼ぶこともある。

フリーソフトは **GNU** のソフトウェアを始めとして、主に **Unix** という研究開発用のコンピュータであるワークステーション用であった。これらのソフトウェアは学術用ネットワークであったインターネット上を中心に発展した。

それに対して、フリーウェア等のオンラインソフトは、個人用コンピュータのパソコン用がほとんどであった。これらのソフトウェアは、商用 **BBS** や草の根 **BBS** といわれた個人やグループが運用するパソコン通信の世界で普及したものである。**1980**

年代半ばから普及し始めたパソコン通信によるネットワークは、アメリカでは **compuserve** や **delphi** などであり、日本においては **NIFTY-SERVE**、**PC-VAN**、日経 **MIX** などがある。オンラインソフトは、これらのネットワークにおいて盛んに公開された。

このようにフリーソフトとフリーウェアは、一般的には違う形態のソフトウェアとして認識されている。主な相違点として、フリーソフトはソースコードを公開することによって、開発の自由や改良の自由を謳っているのに対して、フリーウェアと呼ばれているソフトウェア形態は、ソースコードを公開していることは稀である。

これらの違いは、フリーソフトが **GNU** の理念が強く影響しているのに対して、フリーウェアは個人で作成したソフトウェアを他の人にも利用してもらいたいという善意から自然発生的に誕生したものだからだといえる。

この流れはインターネットの普及によって徐々に違いが薄れてきてはいるが、現在においても顕著である。パソコン用のオンラインソフトの多くはソースコードが公開されていないフリーウェアやシェアウェアが多く、**UNIX** 環境で利用されているソフトウェアはソースが公開されているソフトウェアが多い。このように誕生した環境や状況の違いで独自のコンピュータ文化が形成されている。

2-2-4 オープンソース

オープンソースという用語は1997年に誕生した。当時はインターネットブームと共に **Linux** を活用したビジネスが登場したところであり、ソフトウェアビジネスとフリーソフトウェアとの関係が問われ始めていたころである。

1997年春にフリーソフトウェアの世界を代表する人々が、フリーソフトウェアに懐疑的な人々に、そのアイデアを広める方法を検討した。これらのメンバーには、**Linux** の開発スタイルを広く知らせた「伽藍とバザール」の著者、エリック・レイモンド、**GNU** ソフトの技術書を多く出版しているオライリー社の創設者であるティム・オライリー、当時デビアンプロジェクトのリーダーであったブルース・ペレンスなどである。

かれらは **GNU** の理念はビジネス界に受け入れにくく、そのためフリーソフトウェアの可能性が低くとらえられることを懸念したからだ。これにはいくつかの要因がある。一つは **GNU** のいう”フリー”を無料を意味しており商用ソフトと相容れないと誤解されている。また企業が自社製品にフリーソフトを組み込んで販売する場合、**GPL** が適応されるのを嫌ったためでもある。

オープンソースの元となったのは、デビアン社会契約(**Debian Social Contract**)と

いう **Debian GNU/Linux** の配布方法の記述が元となっている。**Debian** プロジェクトは **Linux** のディストリビューションと呼ばれる配布パッケージの一つである。特徴は、商用ベースではなく非常にオープンな形態でオープンソース・コミュニティによるパッケージ開発を行っている点である。**Linux** 配布パッケージにはいくつか種類があるが、**Debian** は **GNU** の理念に合致したソフトウェアのみを配布パッケージに入れる方針を持っていた。そしてこのガイドラインはフリーソフトウェアとフリーではないソフトウェアを定義するする必要があった。そこでデビアン社会契約を元に、フリーソフトウェアの代わりにオープンソースという概念を提案した。

GNU の主張するフリーソフトウェアとオープンソースの違いは実質的にはない。どちらも同じ形態のソフトウェアを指しているからである。しかし、フリーソフトウェアはプログラムを行う自由、改良をする自由などの理想のための概念であるのに対して、オープンソースはボランティアによるソフトウェア開発の優位性を全面に出している。このように同じソフトウェアの形態を指してはいるが、その考え方がこのように異なっているのである。このようにしてかれらはこれらの考えを「オープンソースの定義」としてまとめた。そしてその考えを広め、また守るためにオープンソース・イニシアティブという団体を設立してオープンソースの定義に適合するライセンスに対して認定を行っている。

2-2-4-1 「伽藍とバザール」とは

オープンソースが注目されるきっかけのひとつに 1997 年に発表されたエリック・S・レイモンドの論文「伽藍とバザール」(**The Cathedral and the Bazaar**)の存在がある。この論文を元にネットスケープ社が自社のWWWブラウザ、ネットスケープナビゲーターをオープンソースにすることを決定したことで有名である。この出来事が企業が戦略としてオープンソース化を推進するきっかけとなった点で、重要な論文である。

この「伽藍とバザール」という奇妙なタイトルは、**Linux** における **OSS** の開発手法について述べたものである。

この論文では、伽藍（原文では **Cahtedral**:大聖堂）というメタファで、大規模かつ複雑なソフトウェアを少数精鋭の開発者が詳細な設計をしながら開発する従来のソフトウェア開発手法を表現している。

それに対して複雑なソフトウェアの代表例でもある **OS (Operating System)** である **Linux** は、多数の開発者が自由に参加しながら開発を続けている。このような従来の常識とは異なるソフトウェア開発手法をバザール方式と表現してその優位性を主張している。

バザール方式の最大の特徴は、そのメタファの通り、多くの開発者が自由に参加し

てくる点である。このメリットをレイモンドは次のように述べている。

ユーザーを持つのはすばらしいことで、それは単に、自分が何かニーズに対応しているんだな、なにか役に立つことをしたんだな、ということを実証してくれるからというだけじゃない。きちんと育てれば、ユーザーは共同開発者になってくれるんだ。(p.78)

つまりユーザーが利用者として意見や機能改善を開発者に伝えるだけでなく、自らも開発に参加することができるような環境の重要性を主張している。このように利用者が同時に開発者というコミュニティによる開発がバザール方式の特徴である。

これによってバザール方式の **OSS** は、多くの人によってチェックされるため、従来の方式よりも素早くバグを減らすことができる。ソフトウェア開発にかかる労力の多くは不具合の修正である。(ブルックス[1975])はプログラミング作業の労力の多くはテストとバグ修正であり、これに対する特効薬はないと述べている。バザール方式がこれらの作業にかかる時間が減少することができるならばそれは画期的である。

またこのようなコミュニティに参加する人たちは強制的ではなく、自発的に参加している。そのため意欲が高く、またスキルも高い。そしてこのような人たちがなぜ **OSS** 開発に携わるのかという動機をレイモンドは次のように述べている。

Linux ハッカーたちが最大化している「効用関数」は、古典経済的なものではなく、自分のエゴの満足とハッカー社会での評判という無形のものだ。(p.99)

この論文で強調されている点は、開発者たちの「効用関数」を最大にする努力が必要である点である。そのため、コミュニティの運用が重要な問題であると主張している。この点は従来の組織と同様である。

このようなコミュニティで開発される **OSS** ではソフトウェアは頻繁に更新される。そして更新されたプログラムを多くの利用者/開発者が利用すると共にその情報がコミュニティにフィードバックされる。その情報を元に新たなソフトウェアが更新されるという循環が生まれている。

しかしこの論文で注意しなければならない点は、全ての **OSS** の開発が、バザール形式で行われているわけではない点である。利用者やテスターは多数であっても開発は少数で行うケースも少なくないことに留意する必要がある。実際、**Apache** や **FreeBSD** など **Linux** と異なる形式で開発が進められている **OSS** プロジェクトも多数存在している。

2-3 オープンソースの特徴

2-3-1 オープンソースの特徴とは

OSS は、これまでのソフトウェア形態と大きく異なる特徴がある。ここでは商用ソフトと比較しながらその特徴を挙げる。

2-3-2 複製・配布の自由

商用ソフトと比較してもっとも異なるのは、**OSS** は複製も配布も自由に行うことのできる権利が与えられている点である。

商用ソフトは、通常、一台のコンピュータにインストールして利用することができる権利をライセンスとして利用者に与えられる形である。この場合、複数台のコンピュータにインストールすることはライセンス違反になる。当然、他人にソフトウェアを複製してそれを渡すことや、ネットワーク上で配布することは禁じられている。

これに対して **OSS** はソフトウェアの複製を推奨し、逆に自由に複製や配布を行う権利をライセンスに盛り込んでいる。例えば **GNU** 一般公有使用許諾書では次のように述べている。

我々は次の2つの方法でユーザの権利を守ります。(1) ソフトウェアに著作権を主張し、(2) 本使用許諾の条項の下でソフトウェアを複製・頒布・変更する権利をユーザに与えます。[オープンソースワールド p.383]

このようにソフトウェアの複製・配布を自由に許可するのは、配布者によって理由が異なる。企業であるならば自社製品の紹介やシェアの獲得など企業の戦略によって許可している場合がある。また、大学や国立の研究所などは税金が使われた開発プロジェクトの成果のため、社会に還元する必要から自由な配布を認めている例もある。さらに上記の **GNU** のようにフリーソフトの理念のために認めているケースもある。

しかし一番の理由は、開発者自身が書いたプログラムを皆に使ってもらいたいという思いである。それは **OSS** の配布ファイルのほとんどは、使ってみた感想を送って下さいという趣旨のメッセージが含まれていることから分かる。

ソフトウェアを自由に複製・配布できることは、そのソフトウェアの発展に大きなメリットがある。ソフトウェアについては元来、開発者と利用者には情報の非対称性がある。開発側はソフトウェアの機能を熟知しているのに対して、利用者は限られた

情報からソフトウェアを選択しなければならない。このため、商用ソフトでは試用版やデモ版を用意しているケースもあるが、入手方法が限定されている。また機能制限がある場合が多く、必ずしも機能していない。そのため最適な製品の選択が困難である。

これに対して **OSS** は自由に製品を複製することができ、全ての機能を試用することができる。また機能に満足すればそのまま利用することもできる。

複製が自由であることは、自由にソフトウェアが流通することを意味している。自分の気に入ったソフトウェアを他人に配ることも可能である。そのためそのソフトウェアを必要とする人に渡りやすくなる。実際、**Web** サイトや **FTP** サイトを通じて多くの **OSS** は流通している。また個人同士でもメールや **FD**、**CD-R** 等のメディアを介してやり取りされている。

このようなコピーフリーのソフトウェアは良いソフトほど口コミで普及をするので、結果として機能的に優秀なソフトほど早く普及する。

そのため **OSS** だけでなく企業が自社製品をフリーウェアとして流通させて普及させる戦略を採るケースも多い。このような例として **Adobe** の **AcrobatReader** やマイクロソフトのインターネットエクスプローラなどがある。これはこのようなソフトウェアの特質を活用した例である。そしてその特質を最大限利用しているのが **OSS** であるといえる。

2-3-3 ソースコード入手の自由

OSS が商用ソフトと異なり、ソースコードの入手する権利を保障している点も **OSS** における大きな特徴の一つである。

ソースコードとはプログラムソースもしくは単にソースとも呼ばれるが、ソフトウェアを作成するためのプログラム言語で書かれたテキスト形式のファイルである。主な言語としては **C**、**C++**、**Java**、**BASIC** などがある。これらの言語で書かれたプログラムは通常、コンパイラと呼ばれるソフトウェアによってバイナリ形式に変換され、コンピュータが実行できる形式のファイルになる。

このようにソースコードは人間が理解できる形式で書かれている反面、実行形式に変換しなければコンピュータに実行させることができない。それに対してコンピュータがソフトウェアを実行するためのバイナリ形式は、人間が解読することは困難である。

そのため単純にソフトウェアを実行するだけであれば、バイナリコードだけで利用できることになり、大多数の商用ソフトはバイナリコードのみ配布されている。これはソースコードがソフトウェアの部品とも設計図ともいえる存在であり、通常は企業

秘密のためである。

OSS がソースコードの入手の自由を強調しているのは、この点に関係がある。**OSS** でのユーザーとは単にプログラムを利用するだけではなくて、そのソフトの仕組みを見ることができるというプログラマーの権利を認めている。

プログラム技術の向上するために最も効果的な方法は、優れた他人のソースコードを読むことである。**OSS** は多くの人が利用することによって洗練されたソースコードの一つである。このような優れたソフトウェアを研究することによって技術の習得が行われ、それをもとに新たなソフトウェアが生まれるという循環が生まれている。

2-3-4 修正・改良の自由

また **OSS** は単にソースコードの入手だけではなく、そのソースコードを元に自由にソフトウェアの修正・改良をする権利があるのも特徴である。

たとえ高価な商用ソフトであっても、ソフトウェアには製品上の欠陥ともいえるバグがある。しかし、購入したユーザーがそのバグを発見しても修正する手段がないのが現状である。通常はユーザーが開発会社に連絡して、修正プログラムが送られてくるのを待つしかないのが現状である。

それに対して、**OSS** にはソースコードを入手することができるので、利用者は開発者の修正プログラムを待つことなく自ら修正することができる。また、修正箇所を開発者に連絡することにより、開発者がプログラム修正にかかる時間を減らすことができる。このように利用者と開発者の協同が行われるのも **OSS** での特徴の一つである。

さらに修正・改良の自由は単にプログラム修正のメリットだけではない。一般に商用のパッケージソフトは既製品であり、全ての人の望んだ機能が搭載されているわけではない。例えば **SI** ベンダーに企業が自社でのソフトウェアの開発を依頼した場合、利用者が望む機能を開発する。ここでは機能はオーダーメイドであり、必要性に応じた機能拡張が行われている。しかし、商用のパッケージソフトは利用者がこのような機能を追加したりすることができない。

OSS ではこのような場合、ソースコードを元に、自らが望む機能を追加することができる。そしてその追加したソースコードを **OSS** の開発に還元することによって、**OSS** が進化していくのである。

実際に、利用者と開発者のこのようなやりとりのなかからコミュニティが誕生することが **OSS** が発展する要因の一つになっている。

2-4 OSS とフリーウェアの違い

ソースコードを公開していないフリーウェアでも、多くの利用者の意見や励ましに

よって支えられている。これもネットワーク上で人々の労力や知識がストックされることによって発展したソフトウェアといえる。

しかし、コミュニティの知識や労力が反映されるものはソフトウェアの機能に対してであり、ソースコードではない。それは開発者がソースコードに機能を反映させていても非公開であるからである。ソースコードが非公開であることは改良や修正が困難なことのみにならず、発展性が低いことが問題である。もしもフリーウェアなどの作者が開発を終了させた場合、その後のメンテナンスを行うことができず発展がそこで止まってしまう。これに対して **OSS** は、ソースコードを公開することと共有することによって、多くの人々の労力や知恵をソースコードとして蓄積している。つまり **OSS** はフリーウェアよりもより広い知識の共有と蓄積が行われている。

ソースコードはソフトウェアという製品を構成する要素である。しかし同時に開発者の労力の成果であり知識の蓄積である。**OSS** のソースコードは、多くの開発者の知識や労力の結晶であるだけではなく、コミュニティや利用者のアイデアや要望などの知識も共有化した成果である。**OSS** はこのような知識の共有により発展しているが、これはオープン・アーキテクチャ戦略でいう情報価値の自己増殖現象の実例として考えることができる。この点が従来型の商用ソフト開発のみならず、フリーウェア・シェアウェアなどのソフトウェア形態と大きく異なる点である。

この違いはフリーウェアなどはあくまでも生産者と消費者が分離した構造なのに対して、**OSS** は生産者でもあり消費者でもある融合した構造になっているのである。その点でフリーウェアやシェアウェアと **OSS** は根本的に異なっている。

また成果の公開と成果の共有を行うことによって、知識の蓄積を行うという側面から **OSS** の開発スタイルを考察すると、その仕組みはアカデミズムのシステムに類似している。

2-5 オープンソースにおけるライセンス形態

2-5-1 オープンソースのライセンス形態とは

オープンソースはソフトウェアライセンスではなく概念である。オープンソースを提唱した人々が設立した **OSI(Open Source Initiative)** では、オープンソースの定義に当てはまるライセンスをオープンソースのライセンスとして認証している。

OSI は、**2001** 年 **11** 月現在、**29** のライセンスについてオープンソースとして認証している。

The GNU General Public License (GPL)
The GNU Library or "Lesser" Public License (LGPL)
The BSD license
The MIT license
The Artistic license
The Mozilla Public License v. 1.0 (MPL)
The Qt Public License (QPL)
The IBM Public License
The MITRE Collaborative Virtual Workspace License (CVW License)
The Ricoh Source Code Public License
The Python license (CNRI Python License)
The Python Software Foundation License
The zlib/ libpng license
The Apache Software License
The Vovida Software License v. 1.0
The Sun Industry Standards Source License (SISSL)
The Intel Open Source License
The Mozilla Public License 1.1 (MPL 1.1)
The Jabber Open Source License
The Nokia Open Source License
The Sleepycat License
The Nethack General Public License
The Common Public License
The Apple Public Source License
The X.Net License
The Sun Public License
The Eiffel Forum License
The W3C License
The Motosoto License
The Open Group Test Suite License

[表 2-2 OSS ライセンス一覧表]

このなかから代表的なライセンスについて説明する。

2-5-2 GPL

GPL(GNU Public License : GNU 一般公的使用許諾) は **FSF** が発表したライセンスである。現在は **1991** 年に発表された **Version 2** になっている。**GPL** は **FSF** が提唱するフリーソフトウェアを維持するために法律的側面から保証するためのものである。

ソフトウェアで最も自由なライセンス形態は著作権を放棄した **PDS(Public Domain Software)** といえる。しかし著作権を放棄したソフトウェアは、誰かが著作権を主張することが可能であり、**FSF** のいう独占的なソフトウェアになる可能性がある。そのため **GPL** によって著作権を主張してフリーソフトウェアを維持しているのである。**GNU** プロジェクトではこの考えを次のように述べている。

独占的なソフトウェア開発者たちは、ユーザの自由を奪うために著作権を使います。私たちは、ユーザの自由を保障するために著作権を使います。これが、私たちが自分たちの主張を著作権、すなわち「コピーライト」をもじって「コピーレフト」と名付けた理由です。¹

GPL の特徴はいくつかの条件を満たしている場合には、ソフトウェアの複製・配布は自由であるとしている点である。その主な条件のなかでもっとも特徴的な点は、**GPL** でライセンスされたソフトウェアを改良や修正をして再配布する場合は、そのソフトウェアも **GPL** でライセンスしなければならない点である。

例えば **GPL** でライセンスされたソフトウェアを企業が改良して販売しようとしても、そのソフトは **GPL** を適用しなければならないことになる。これは **GPL** でライセンスされたソフトウェアが、**GPL** 以外のライセンスにならないような工夫である。

このように **GPL** はユーザーの自由の権利を強力に守るライセンスである。しかしこのことは企業が **GPL** でライセンスされたソフトウェアを利用することを難しくしているともいえる。

2-5-3 X,MIT,BSD

X Window System や **MIT** 形式のライセンス、**BSD** ライセンスなどは **GPL** について広く使われているオープンソースのライセンス形態である²。

1984年に公開された **X Window System** は、**Athena** プロジェクトの一環として **MIT** (マサチューセッツ工科大学) コンピュータサイエンス研究所と **DEC** の共同研究で誕生したシステムである。ここで開発された **X Window System** のためのライセンスを **X** スタイルのライセンスと呼んでいる。

また **MIT** 形式のライセンスは、**MIT** で開発・配布されるソフトウェアに使われているライセンス形式を指す。同様に **BSD(Berkeley Software Distribution)** とはカリフォルニア大学バークレイ校が配布するソフトウェアに付けられるライセンス形態

¹ <http://www.gnu.org/copyleft/copyleft.ja.html>

² <http://www.jp.freebsd.org/www.freebsd.org/ja/copyright/license.html>

である。従来の **BSD** ライセンスでは **BSD** ライセンスのソフトを組み込む場合、カリフォルニア大学バークレイ校で開発されたものであるといった趣旨の告知が義務づけられていた。1999年に撤廃されそれ以降の **BSD** ライセンスは **X** や **MIT** 形式のライセンス形態とほぼ同等になっている。

しかし一般に **BSD** スタイルのライセンスといわれる場合は、従来型の **BSD** ライセンスを指している例が多い。そして、これらのライセンス形態は上記の組織以外で作成されたソフトウェアに多く流用されている。

GPL と上記のライセンス形態の最大の違いはライセンスの継承が求められていない点である。これは、これらのライセンスされたソフトウェアを改良・修正した場合、元のライセンス形態以外のライセンスを適用しても良いことを意味する。つまりこれらのソフトウェアを商用ソフトに取り込むことも許可されていることを意味している。実際に商用 **X Window System** や商用の **BSD** 系 **UNIX** である **BSD/OS** など、多くの商用ソフトウェアが存在している。

また再配布者にソースコードの提供を求めている点も **GPL** と大きく異なる点である。つまり **X,MIT,BSD** ライセンスなどは、ユーザーの自由を主眼においた **GPL** と違い、成果物の公開と促進に重点がおかれているのである。これはいずれも大学研究機関が研究・開発した成果物を公開することを目的としているためである。

2-5-4 NPL

NPL(Netscape Public License)はネットスケープ社の **WWW** ブラウザである **Netscape Navigator** のソースコードをオープンソース化するためのライセンスである。これは企業が自社の商用ソフトをオープン化することによって、ユーザーコミュニティと共に開発を進めるという意味で革新的である。

NPL の特徴は、**GPL** と **BSD** スタイルを併せ持った形になっている点である。これはネットスケープ社のみが **BSD** スタイルの形式、特に非公開の派生プログラムの作成を許可する特権を持つが、それ以外は **GPL** スタイルのライセンスにするというものである。これにより開発元であるネットスケープ社のみが商用のライセンス形式で製品化することができ、それ以外はフリーのソフトウェアとして利用されるという、利用者と開発元の双方に利点のあるライセンスになっている。

その後ネットスケープ社は **NPL** で保護されているソースコード以外に新規で開発されたソースコードは **Mozilla Public License(MozPL,MPL)** という **NPL** からネットスケープ社に与えられた特権を除いた形のライセンスを適用することとしている。これによって例えば、ユーザーが新規に開発したソースコードがネットスケープ社の独

占になることを避けられるようになった。

現在このような企業によるオープンソース化のためのライセンス形態は、**IBM**、**Apple** と多くの企業が適用し始めている。

第3章 OSS 開発における原動力の分析

3-1 分析の手法

ネットワーク上で **OSS** の開発に関与している集団は、一般にオープンソース・コミュニティと呼ばれている。たとえば **Linux** に関心のあるグループは **Linux** コミュニティと呼ばれており、また自称している。コミュニティとはアメリカの社会学者マッキーバーが設定した概念である。しかし現在、コミュニティという概念は大変幅広く使われて一概に定義することは困難である。

ネットワーク上のコミュニティについて根来(1999)らは「あるバリューに関心を持つアクセス者同士が、商取引と独立して行う知識交換活動」を知識インタラクションと定義し、「知識インタラクションで交換された知識の意図的な蓄積」を知識コミュニティと呼んでいる。そして **Linux** コミュニティのような参加資格を限定しない開放的な形態をオープン・コミュニティと定義している。

また國領(1999)はコンピュータネットワーク上で共通の目標に向けて協働構造をつくれやすいことを挙げて、コミュニティを「ある価値体系を共有し、その価値観の下に協力する関係を持っている集団」と位置づけている。

本論文でもコミュニティはそのような文脈で使用する。つまりオープンソースにおけるコミュニティはオープンソースという概念、もしくはオープンソースという財を共有してその価値観の下に開発や活用について協力する関係を持っている集団であるとする。

オープンソースについての議論の多くは、コミュニティが **OSS** 開発の原動力であることを指摘している。しかし **OSS** コミュニティのように、価値体系を共有してその価値観に基づいて協力する関係を持つ集団は新しいものではない。コミュニティだけがオープンソースの原動力ならば、**OSS** が以前から存在していてもおかしくないのではないだろうか。実際は、**OSS** の運動が流れが盛んになったのが 90 年代後半になってからである。**OSS** の原動力はコミュニティの力であると単純化した議論ではなく、より広い視野で他の要因の存在を考える必要がある。

本章では **OSS** 開発の原動力は **OSS** コミュニティだけではなく、情報化、デジタル化、ボランティアなど他の要因と深く結びついていると考えている。そこで文献調査を元に **OSS** 開発の原動力をデジタル財、ネットワーク、ボランティアという 3 つの側面から分析を行う。

3-2 デジタル財

3-2-1 デジタル財とはなにか

今日、デジタル財や情報財という概念が注目を浴びている。しかし概念自体は以前から議論されてきたものである。

例えば(松岡[1986])らは、情報財の性質として次の3点を挙げている。

- 1：使用保存性（非消耗性・耐久性）
- 2：同時流通性（非転移性・協働性）
- 3：相互付加性（累積効果性・添加性）

[情報と文化 P35-P36]

この中の使用保存性とは、情報がいくら使用されてもそれ自体は消耗しない性質を現わしている。そして同時流通性とは、情報がAからBへ転移されたり流通されても情報はAの手元にも残ることを意味している。ソフトウェアの側面から考えれば、複製が容易に可能であることといえる。最後に相互付加性、もしくは累積効果性とは、情報の上に新たな情報を加えていくと情報の価値が増殖しうる点を現わしている。

情報財が重要な概念になっているのは情報財のデジタル化が近年、急激に進行したためである。デジタル財とは、これまで情報財と呼ばれていた財がデジタル化されたものを指している。従来の情報財とデジタル化された情報財、つまりデジタル財の違いは、情報が物理的媒体の束縛から乖離することが可能か否かである。

このような現象をMITメディアラボのニコラス・ネグロポンテは、「アトムからビット」と表現している。これはアトム、つまり原子で構成された物理的なものからビット、デジタル化されたモノへの価値が増していることを指している。

このような情報のデジタル化について、(奥野・中泉[2001])らはソフトウェアやCD等のデジタル化された楽曲のように、媒体から情報をアンバンドル可能にしていく現象をデジタル化と呼び、このような財をデジタル財と定義している。また(國領[1999])はデジタル財を電子的に流れ、経済価値を持つデジタル符号の集合としている。

3-2-2 デジタル財の公共財的側面

デジタル財の大きな性質は、複製コストがきわめて低いことである。従来の情報財は、物理的媒体から純粋に切り離されることはできなかった。例えば情報財としての書籍を複製するためには、コピー機などで複写することはできたが、複写するコストと複写による情報の劣化は避けられなかった。レコードなどの音楽も同様である。レコードからカセットテープに複製することは可能であっても、それは完全なコピーに

はならず、オリジナルに対して劣化した情報を持つコピーでしかなかった。

ところがデジタル化された情報は、きわめて簡単に複製が可能であるだけでなく、オリジナルと完全に同一のコピーが可能である。つまり自由に複製が可能であるならば、どれだけ大人数であっても全くオリジナルと同じものが利用できる。そして一度、ネットワーク上に流通したデジタル財は回収不能になってしまうのである。このようにデジタル財は公共財的側面があることがわかる。

ところがデジタル財のこのような性質は、従来の情報財にとって大きな問題を引き起こしている。情報財のデジタル化によって、著作権や特許権で守ることができなくなったことである。現在、インターネット上には、パッケージソフトやCDの楽曲、DVDの映像などがアトムの媒体からビットへ変換され、デジタル財として無断に流通しているのが実状である。実際のところ、これらの問題に対する有効な手段はないように思われる。そしてその理由がデジタル財の本質的性質のためではないだろうか。

3-2-3 OSS との関係

このように情報財のデジタル化は、従来の制度を大きく揺るがしている。しかしOSSは逆にデジタル財の性質を最大限に利用しているといえる。(國領[1999])はデジタル財のなかで無償で流通しているものを無償デジタル財と呼んでいる。そして無償デジタル財が存在する要因として複製・流通時の変動費の低いことを挙げた。この性質は無償提供の容易さと同時に課金コストの高さを意味しており、無償配布がしやすいデジタル財に対する課金は困難であることを述べている。

OSSはこの無償デジタル財に分類される性質のものである。この特徴がOSSの普及を可能にしている。情報財の相互付加性によって、OSSは多くの人の手によって改良が進められ価値を高めているのである。

このようにデジタル財は本来、情報財が持っていた性質をよりはっきりさせたものではないだろうか。そして従来の情報財であれば不利になる環境において、OSSは逆にデジタル環境に適応したデジタル財なのである。

3-3 ネットワーク

3-3-1 二つの「ネットワーク」

OSSはネットワークの発展と共に普及している。ここでいう「ネットワーク」には二つの意味がある。それはソフトウェアをデジタル財として流通させるためのコンピュータネットワークと、人と人をむすびつける意味でのコミュニケーションとしてのネットワークである。ここではその二つの「ネットワーク」とオープンソースの関係

について考察する。

3-3-2 コンピュータとしてのネットワーク

3-3-2-1 1980 年前半までの状況

コンピュータネットワークは **1960** 年代から存在していた。例えば銀行や鉄道のオンラインシステムは、コンピュータネットワークの一つである。しかしこれはある特定の目的のために構築された、いわば閉じられたネットワークである。これらのネットワークは、自由にデジタル財を流通させることはできないシステムである。それに対して今日のインターネットは、だれでも自由に接続することのできる開かれたネットワークである。

実際、パソコン通信や自由に利用できるインターネットなどが広く普及していなかった時代は、物理媒体によってフリーソフトウェアは配布されていた。たとえば **GNU** の **emacs** や初期の **BSD** は、磁気テープの郵送で配布されていた。またパソコン用のフリーウェアもパソコン通信が一般化されるまでは、カセットテープや5インチフロッピーディスクで配布されていた。さらに以前は雑誌記事として掲載されたソースコードを利用者が入力することで配布されることさえあった。

3-3-2-2 開かれたネットワークへ

インターネットやパソコン通信が普及し始めたのは **1980** 年代半ばからである。当時、インターネットの利用は一部の大学研究機関に限られていて、今日のようにだれでも利用できるネットワークではなかった。またパソコン通信は、アメリカでは **1970** 年代後半に商用のパソコン通信が誕生した。そして日本のパソコン通信は電気通信事業法が改正によって回線自由化が行われた **1985** 年に誕生した。またこの回線自由化によって日本でも **JUNET(Japan University Network)** が開始され、日本でもインターネットが開始された。

これらのネットワークは現在のインターネットの規模と比較すると、小規模なネットワークではあった。しかし利用者が自由に使うことのできるネットワークが誕生したことは、ソフトウェアを物理的な財ではなく、デジタル財としてネットワークを利用して配布することが可能になったことを意味している。実際、フリーソフトという概念を生み出した **GNU** プロジェクトが開始されたのが **1983** 年であり、パソコン用オンラインソフトであるフリーウェアと呼ばれるソフトウェアの普及もパソコン通信の普及と同じくしている。

そして **1990** 年代にインターネットの商用利用が認められると、インターネットの利用は爆発的に増加した。たとえば **Linux** の最初のバージョンが公開されたのは

1991年である。オープンソース運動が盛んになったのはこの時期からである。

このように開かれたネットワークの存在がフリーウェアやオープンソースを生み出す環境として必要なことが分かる。これはだれでもオープンに利用できるからこそ、自由にデジタル財を流通されることが可能になるからである。そして開かれたネットワークの存在こそ、デジタル財の公共財的側面を促進させる環境を生み出しているのである。

3-3-3 コミュニケーションとしてのネットワーク

3-3-3-1 人と人をつなぐネットワーク

パソコン通信やインターネットなどは、コンピュータネットワークであると同時に人と人を結びつけるコミュニケーションとしてのネットワークでもある。開かれたコンピュータネットワークは、時間的・距離的制約を越えて人々の出会いとつながりを作り出している。これによって主義や主張、趣味などさまざまな共通の価値観を持つ人々が、ネットワーク上にコミュニティを作り出した。**OSS**もそのなかの一つである。

3-3-3-2 OSS コミュニティの形成

OSSにとってネットワークの重要性はソフトウェアの配布だけではない。ネットワークは、ソフトウェアの利用者と利用者、開発者と利用者といった、これまではあまり接触がない状態であった関係を、相互に結びつける環境を作り出している。

今日、多くの**OSS**にはサポート **Web** サイトがあり、コミュニティが存在している。これらコミュニティはネットワーク環境と共に進化している。コミュニティは電子掲示板、メーリングリスト、**IRC(Internet Relay Chat)**、ニュースグループなどのシステムを利用することで、コミュニティを成長させている。そしてコミュニティが新たなコミュニケーションツールを創造している。

これらネットワーク上で誕生したコミュニティは、利用者同士、利用者と開発者との間で発生するインタラクションのなかで、ソフトウェアに大きな影響を与えている。

たとえば利用者が作るソフトウェアの紹介 **Web** サイトには、多くのリンクが張っており、関心がある人を次々に結びつける役割を果たしている。また、このようなサイトを閲覧したものが別の人に紹介し合うことにより、口コミでそのソフトウェアが多くの人に知られ、普及を加速させる効果がある。

また利用者の率直な意見や、利用者同士のコミュニケーションの中で生まれるアイデアは、開発者にとって貴重な情報である。利用者同士だけではなく、利用者が開発者に挙げるバグ報告などの連絡は、ソフトウェア品質を素早く向上させている。

特に **OSS** ではこのようなコミュニティの存在は重要かつ、活発に行われている。

OSS の場合、利用者と開発者の関係は密接で、利用者がバグの修正を行うことがあるなど開発者と利用者の敷居は他のソフトウェア形態よりも低い。

このように新たなネットワーク環境が新たなコミュニティを生み出し、**OSS** はその環境にうまく適用している。特に **OSS** にとってネットワーク上のコミュニティは必要不可欠なものといえる。

このようにコミュニケーションとしてのネットワークは、いままでにない出会いとつながりを生み出すことで新たな集団を生み出している。これらはオープンソース開発コミュニティもそのなかの一つの実例だといえる。

3-4 ボランティア

3-4-1 ボランティアとは

ボランティアという概念は、インターネットコミュニティだけではなく、**NPO** や社会的なボランティア活動まで多くの集団について言及する上で重要なキーワードになっている。ボランティアとは「自発的な」「自由意志からでた」などを意味している。

ボランティアはインターネットコミュニティの原動力としてだけではなく、企業や大学など **OSS** に関わる集団全体に共通するキーワードのひとつになっている。ここではボランティアと **OSS** についての関係について述べる。

3-4-2 OSS コミュニティとボランティア

OSS コミュニティに対するもっとも大きな疑問はその原動力とはどのようなものなのかという点である。例えば(木村ら[2000])は **OSS** を中心にボランティア経済と呼ばれる構造が生まれており、**OSS** コミュニティの参加者は既存のボランティア行為と比べて利己的・自己志向的であることを指摘している。また(ヒマネン[1999])は、アカデミック・モデルとオープンソース方式はどちらも個人の自発的な情熱に基づいて行動していることや、仲間の賞賛に動機づけされている点を指摘して、オープンソース方式とアカデミズムの類似性について述べている。

どちらの主張に共通していることは、個人が利己的な目的であれ、情熱や仲間の賞賛などの動機であれ自発的に **OSS** に関わっている点である。

3-4-3 ボランティアコモンズ

ボランティア・コモンズとは(金子・松岡・下河辺[1998])らが提唱した概念である。

金子(1998)はボランティア・コモンズについて次のように述べている。

ボランティア・コモンズの特徴は、自発性、情報共有、関係変化、相互性、フラジリティ (=危うさ) などである。もう少し具体的にいおう。まず、メンバーは自発的に参加する。メンバーそれぞれが情報を自発的に供出し、全体として情報を共有する。メンバー間の既存の関係が変化することでコモンズが成り立っている。そして、コモンズで共有された情報や経験が蓄積された「共同知 (=コミュニティナレッジ)」をメンバーが利用することでそれぞれのメンバーにとって何らかの具体的なメリットが存在する。そのことがコモンズを維持する動機となっている。
(p.7)

これは **NPO** やインターネット上の組織など幅広い組織に現れる現象を指しているもので、オープンソースに特化したものではない。しかし **OSS** コミュニティを広くインターネット上の組織として捉えると、この概念は **OSS** コミュニティを的確に表しているといえる。

OSS コミュニティをボランティア・コモンズの視点から考察してみると、次のような説明ができる。**OSS** は、自発的にコミュニティに参加した開発者たちによって開発されている。また利用者が開発者として参加することや、バグ報告や改善項目の提案などの協力を自発的に行っている。このように参加者は自発的に知識や情報をコミュニティに提供しあっている。これは上記で述べられているコミュニティナレッジにあたる。そして開発者はコミュニティに参加することで技術力向上や名声を得ることを動機とし、利用者は高品質のソフトウェアを無償で利用することができる。

このようにコミュニティのメンバーがインターネット上で知識共有し、その知識を発展的に蓄積させていく。このような組織をボランティア・コモンズと呼ぶのであれば、**OSS** コミュニティはまさしくその実例といえる。

ボランティア・コモンズはネットワーク社会において現れる、従来の経済的利益とは異なるメリットで結ばれた組織構造について述べている。**OSS** コミュニティはこの組織構造の典型ではないだろうか。

3-4-4 企業戦略としてのボランティア

そしてネットワーク上でボランティア的な行為を行うのは、個人だけではなく企業でも行われようとしている。例えば、オープン・アーキテクチャ戦略とは(國領[1999])が提唱しているビジネス戦略である。國領はこの戦略について次のように定義している。

自社の持つ情報をより積極的に公開、発信し、他者の多様な情報と結合させることによって、情報価値の自己増殖現象を発生させ、その価値を自社の利益として取り込んでいく戦略のことである。(p.1)

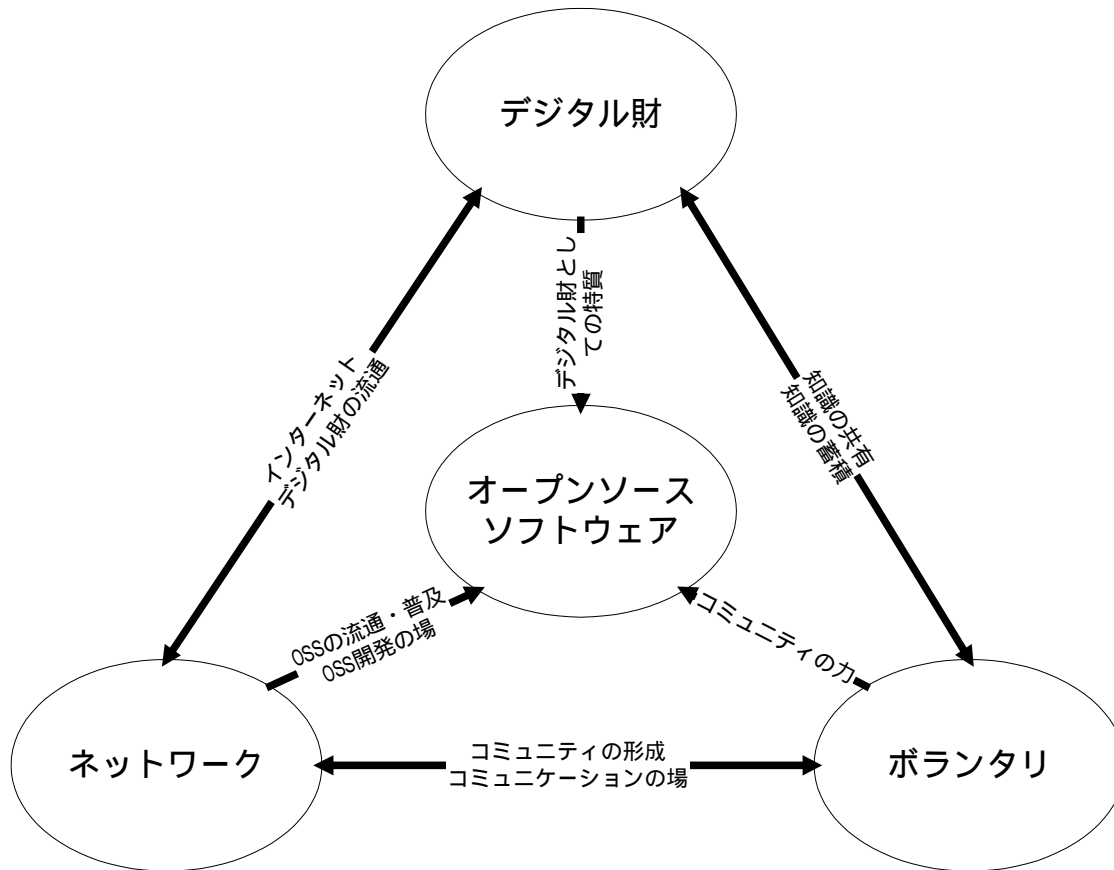
一般的に考える企業とは、自社の資産や情報を増やして囲い込むことによって利潤を得る戦略である。これに対してオープン・アーキテクチャ戦略は逆である。お互いの知識を出し合って共有することによって自社の利益を得るという考えである。またこの戦略が情報技術と深く関わっていることを指摘している。

3-5 分析結果

分析結果をまとめると以下の3点になる。

1. **OSS** のデジタル財としての性質は、ネットワークを使って自由に流通することを可能としている。同時にネットワークを介してコミュニティがソースコードの共有や知識の蓄積を可能にしている。このように **OSS** は従来のソフトウェア配布形式よりもデジタル財としての性質を活用している。
2. ネットワークは **OSS** の流通機能として作用している。同時に **OSS** はインターネットを構築するシステムを創り出している。また **OSS** を中心に自発的に集まる人々を結びつける役割も果たしている。そしてネットワーク上にコミュニケーションの場を造りあげることによって **OSS** コミュニティが形成されている。
3. **OSS** にボランティアに関わる個人・集団がネットワーク上でコミュニティを形成している。また知識や労力をデジタル財として共有や蓄積しながら発展している。

このような考察から **OSS** の原動力は、**OSS** コミュニティというよりは、むしろ図のように **OSS** を取り巻く環境であると捉えるべきではないだろうか。



[図 3-1 OSS における原動力]

OSS はコミュニティによって開発されているという意味では、コミュニティが **OSS** 開発の原動力と呼ぶことは間違いではない。しかし、**OSS** という製品を取り巻く要素はソフトウェアのデジタル財としての性質でありネットワークの存在であり、コミュニティだけではなく企業や大学等を含めたボランタリーな行動である。**OSS** の原動力はこれら **3** つの要素が関わり合う環境が作用している。

第4章 大学と OSS に関する調査

4-1 調査方法

本章では大学と OSS との関係を調査する。調査内容は次の 2 点である。OSS 開発に関わる人々がどのような組織に属しているかを、Linux ディストリビューションから調査する。次に大学発の OSS がどのように発展したかを事例を元に調査する。調査した OSS は PostgreSQL という RDBMS である。

4-2 ディストリビューション調査

4-2-1 調査内容

本研究ではオープンソース開発プロジェクト参加者の所属内訳について調査した。これは、オープンソース開発プロジェクト参加者が、どのような組織の立場から参加しているかを調査するのが目的である。

オープンソース開発の参加者はその多くが自主的に参加している。しかし参加者は企業なのか個人なのか、また大学研究機関に所属しているのか、客観的な資料が存在していないのが現状である。そこで本研究では Linux ディストリビューションに含まれるメールアドレスを抽出することにより、オープンソース開発プロジェクト参加する人々の基礎的属性を調査した。

4-2-2 調査対象

Linux ディストリビューションとは、**Linux** カーネルとその上で動作するアプリケーションソフト群をユーザーが利用しやすいようにパッケージ化したものである。例えるならば、マイクロソフト社の **Windows** などの商用 **OS** と、その上で動作する **MS-Office** などのアプリケーションソフトをセットにしたものに近い。

Linux ディストリビューションは複数の企業・団体が配布している。

ディストリビューション名	販売会社・団体	系統
Kondara MNU/ Linux	デジタルファクトリ株式会社	RPM 系
LASER5 Linux	レーザーファイブ株式会社	RPM 系
Linux Mandrake	MandrakeSoft S.A	RPM 系
Linux MLD	メディアラボ株式会社	RPM 系
LiveLinux	メディアラボ株式会社	RPM 系
Miracle Linux	ミラクル・リナックス株式会社	RPM 系
HOLON Linux	株式会社ホロン	RPM 系
OpenLinux	Caldera International, Inc	RPM 系
Red Hat Linux	レッドハット株式会社	RPM 系
SuSE	SuSE, Inc.	RPM 系
Turbolinux	ターボリナックス ジャパン社	RPM 系
Vine Linux	Vine Project	RPM 系
Plamo Linux	個人	TGZ 系
Slackware	The Slackware Linux Project	TGZ 系
Debian GNU/ Linux	Debian	DEB 系
Omoikane GNU/ Linux	オモイカネ株式会社	DEB 系
プロサーバ for Linux	富士マグネディスク株式会社	DEB 系

[表 4-1 主な **Linux** ディストリビューション一覧]

通常、企業が配布しているディストリビューションは有償バージョンと無償バージョンがある。これは **Linux** システムの多くのソフトウェアが **GPL** でライセンスされており、ソースコードの公開が必要であるためである。それに対して一般の商用ソフトウェアはバイナリ配布であり有償である。従って **Linux** ディストリビューターは、商用ソフトウェアを含んだバージョンを有償で販売し、含まないバージョンを無償でインターネット上や雑誌添付の形で配布している。また一般に有償バージョンを購入すると、無償バージョンにはないディストリビューターのユーザーサポートを受けることができる。このようにユーザーは商用ソフトの必要性やサポートの有無に応じて、パッケージを選択することが可能である。

今回、調査した **Linux** ディストリビューションは **Vine Project** の **Vine Linux2.1.5FTP** 版である。**Vine Linux** は **RedHat** 社の **RedHat Linux** を元に日本語

環境の整備など日本人向けにカスタマイズしたディストリビューションである。

Linux ディストリビューションを調査対象にした理由としては以下の点である。

1) **Linux** が現在最も普及しているオープンソースによるシステムである

2) **Linux** 上のアプリケーションソフトウェアは **Linux** のために開発されたわけではなく多くの **OS** に移植されている。従って **Linux** ディストリビューション上のアプリケーションソフトを調査することは、一般に普及している **OSS** の多くを網羅することができる

3) 一部の関係者に対するアンケート調査は異なり、ソースコードやマニュアルといった実際の成果物から情報を抽出することができる

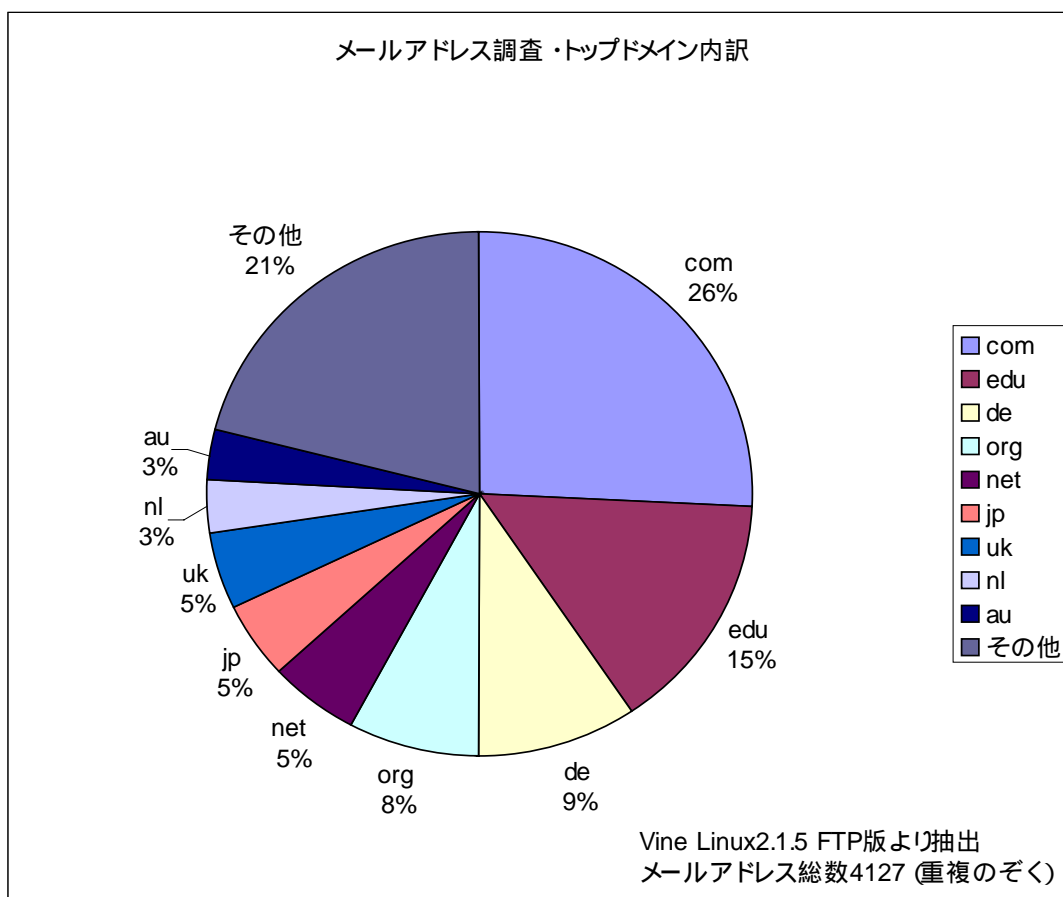
4-2-3 調査方法

調査方法として **Linux** ディストリビューションに含まれる **OSS** のソースコード・マニュアル類からメールアドレスを抽出して、そのデータを分類する。

具体的手順としては、**Vine Linux2.1.5** のソースコードが収録されているソースパッケージ(拡張子 **SRPM**)を全て展開した。ファイル総数は約 **12** 万ファイル、ファイル容量は約 **1.2GB** ほどであった。この中から今回は **C** 言語のソースコード拡張子である **.c** のファイルを抽出した。そして展開した全てのソースコードからメールアドレスを取り出した。メールアドレスの抽出方法は、抽出用プログラムを作成して、それを用いて行った。

4-2-4 調査結果

4-2-4-1 トップドメイン



[図 4-1 メールアドレス調査・トップドメイン内訳]

ドメイン名称	意味
com	商業的ビジネス、または企業
edu	北米教育機関
de	ドイツ
org	非営利的活動をする機関
net	ネットワーク団体
jp	日本
uk	イギリス
nl	オランダ
au	オーストラリア

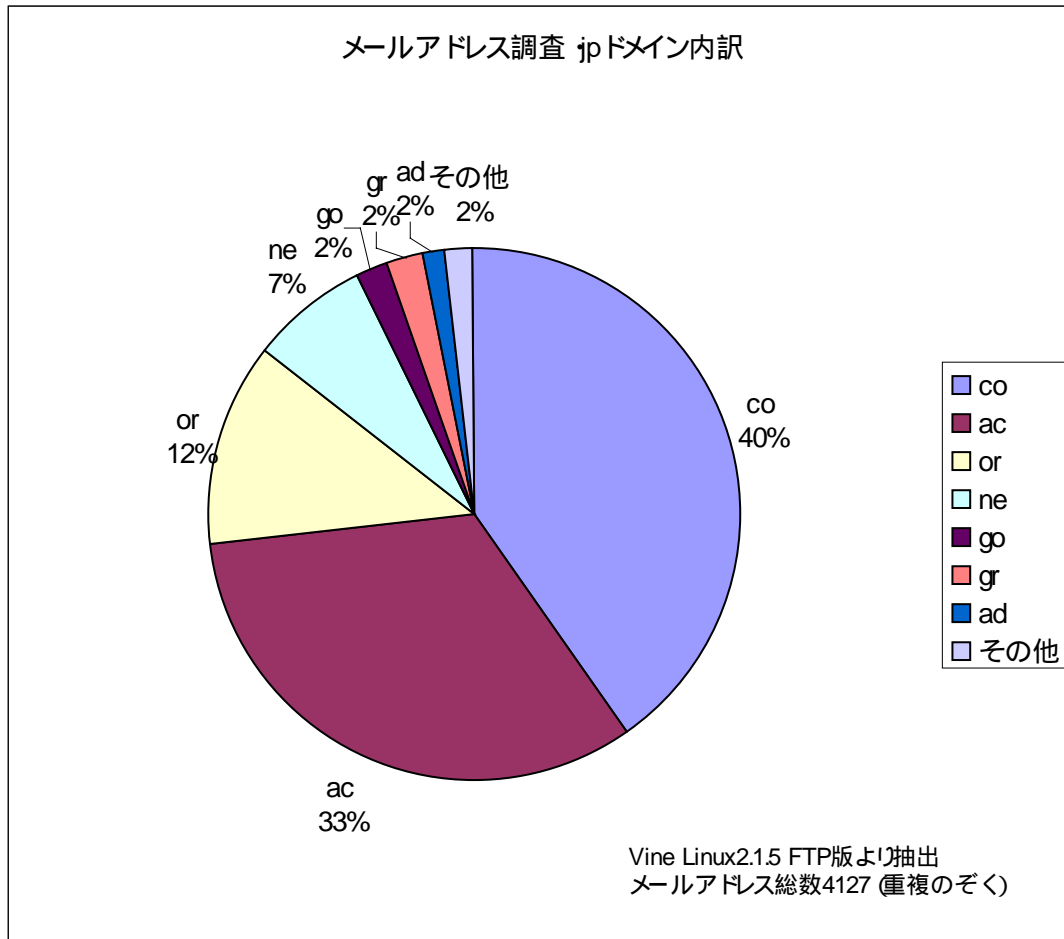
[表 4-1 図 4-1 におけるドメイン名称]

VineLinux のソースコードに含まれるメールアドレスのうち、**com,edu,de,org** で全体の約半数を占めている。**com** ドメインは企業などの商用で利用されるドメインである。このドメインに含まれているのは所属する企業の社員である場合と、**AOL** や **NIFTY** など **ISP** の会員で有る場合が考えられる。したがって **com** ドメインに含まれるメールアドレスの全てが企業に所属しているとは限らない点に留意する必要がある。

次に 2 番目に占めているトップドメインは **edu** ドメインである。このドメインは北米の教育機関が利用するドメインであり、そのほとんどが大学である。そのためこのドメインのメールアドレスを利用している者は、大学等の教育機関に所属していると考えられる。

また 4 番目に占めている **org** ドメインは非営利の団体が利用するドメインである。**OSS** コミュニティの多くが **org** ドメインを利用している。例えば **GNU** プロジェクトなら **gnu.org** であり、リナックスならば **linux.org** を使用している。グラフに現れている **org** ドメインアドレスの多くがこのような **OSS** コミュニティが利用しているドメインである。

4-2-4-2 jp ドメイン



[図 4-2 メールアドレス調査 jp ドメイン内訳]

ドメイン名称	意味
co	企業
ac	学術組織
or	法人格をもった団体
ne	ネットワーク運用組織
go	政府機関
gr	法人格を持たない団体
ad	JPNIC 会員

[表 4-2 図 4-2 におけるドメイン名称]

ここでは日本が利用している国別トップドメインである **jp** のうち、さらに第 2 ドメインを展開してグラフ化したものである。

jp ドメインのうち最も多いのは **co** ドメインで、全体の 4 割を占めている。**co** ドメインは企業が利用するドメインである。トップドメインの場合と異なり、**ISP** の会員

が利用するドメインは **ne.jp** もしくは **or.jp** であり、この **co** ドメインのメールアドレスを利用して利用している者は企業に属していると考えることが可能である。

jp ドメインの内訳で次に多いドメインは **ac** ドメインである。**ac** ドメインは学術組織が利用するドメインでおもに大学が利用している。**ac** ドメインは全体の **33** パーセントを占めている。

このように **jp** ドメインに含まれる第 **2** ドメインの分布は企業と大学で全体の **7** 割弱をしめていることが分かった。

4-2-5 分析

4-2-5-1 OSS の開発メンバーには大学関係者が多い

Linux ディストリビューションの調査ではディストリビューションに含まれるメールアドレスのうち、**15** パーセントが北米の教育機関のドメインであった。またそれ以外の国の教育機関も含めるとさらに多くの人々が教育機関に所属していると考えられる。

このことは **OSS** 開発参加者が大学などの教育機関以外の組織、もしくは個人が多く開発に関わっていることが分かる。これは **OSS** に関わる集団に多様性があることを意味している。ここにはユーザーとして個人が参加している場合もあれば、企業として関わっている可能性などが挙げられる。このようにメールアドレスの分布から、**OSS** を軸とした多様な組織の集まりが存在することを確認することができた。

4-2-5-2 日本は学生の比率が高い

また日本のドメイン (**jp**) の内訳では教育機関のドメインである **ac.jp** が全体の約 **30** パーセントを占めている。このように日本における大学機関関係者による **OSS** 開発参加者はアメリカと比較して高い。本調査では実際に大学関係者の比率が高い点を客観的資料を基に明らかにした。これは、日本の大学は **OSS** を開発する能力を持つ人的資源を持っていることを明らかにできた。

4-3 事例研究

4-3-1 調査目的・方法

オープンソースが誕生するためには最初のトリガーが必要である。つまり誰かが最初のプログラムをコーディングしてオープンソースとして公開する必要がある。このような **OSSS** のトリガーとなるケースとしては、個人のケース、大学・研究機関の研究成果がトリガーとなるケース、企業の製品がトリガーとなるケース、初めから **OSS** 開発プロジェクトで始まるケースに分類することができる。

開発主体	代表例
個人	Linux., Perl, Namazu, emacs
大学または研究機関	Xwindow, postgresSQL, Apache
企業	Canna, Mozilla, mySQL

[表 4-3 OSS 開発主体]

本論文で注目するのは大学研究機関の研究成果がトリガーとなるケースである。これは上記の表でも分かるように、大学研究機関の成果がトリガーになった **OSS** は大規模なケースが多く、社会に対する重要度が高いと思われる。また個人がトリガーとなったケースであっても **Linux** など開発者が学生時に開発が始まったケースも多く、これらも広義として大学研究機関がトリガーとなったケースに含めると **OSS** の多くが大学という組織となんらかの関係がある。

本節では大学研究機関が **OSS** に果たす役割を調査することを目的とする。大学研究機関がトリガーとなった **OSS** がどのように発展しているかについて事例研究を行う。本節では、代表的な事例としてオープンソースの **RDBMS** である **PostgreSQL** を取り上げる。

調査方法としては文献調査とインターネット上の開発コミュニティサイトの情報、さらに **PostgreSQL** ソースコードから調査を行う。

4-3-2 PostgreSQL とはなにか

4-3-2-1 RDBMS とは

PostgreSQL はオープンソースの **RDBMS**(**Relational Database Management System**)である。**RDBMS** とは関係データモデルと呼ばれるデータを表の形で現わすデータベースである。**RDBMS** は大量のデータを蓄積・検索するために用いられるシ

システムである。通常、**RDBMS** は **SQL (Structured Query Language)** を用いてデータに対する操作を行う。

大規模 **DB** に利用される商用ソフトウェアではオラクルの **Oracle9**、マイクロソフトの **SQL Server**、**IBM** の **DB2** などがある。これらは企業や組織などのデータを扱う基幹システムである。これらの **RDBMS** は経理データや顧客データ、また大量の実験データ等処理しており、ミッションクリティカルな用途に用いられている。このため非常に高い信頼性が求められるソフトウェアである。

またオープンソースの **RDBMS** としては **PostgreSQL** のほか、ポーランド社の **InterBase**、**TcX DataKonsalt** 社の **MySQL** などがある。**InterBase** は商用版とオープンソース版があり、**MySQL** は独自の商用ライセンスと **GPL** ライセンスの併用になっている。**MySQL** と **InterBase** は企業の製品であったものをオープンソース化する事で新しいビジネスモデルを構築している。

4-3-2-2 PostgreSQL の特徴

PostgreSQL は次のような特徴がある。

SQL92 のサポート

SQL は **RDBMS** を操作するときに使われる問い合わせ言語であり、**PostgreSQL** は標準的な **ISO** や **JIS** で標準化された **SQL92** の機能の多くをサポートしている。

多くのプラットフォームに対応

Linux, FreeBSD, NetBSD, OpenBSD, Solaris(Sparc, x86), HP-UX, Windows など多様なプラットフォームに移植されている

多様なプログラミングインターフェースを持つ

perl, JAVA, Ruby, C++, Delphi, Python など多くのプログラムから **PostgreSQL** を利用することができる

豊富なデータ型を持つ

一般的な **RDBMS** 以外のデータ型として、円や多角形を格納する独自のデータ型や、配列型やユーザーが定義できるデータ型などのオブジェクト志向のデータ型をもっている

4-3-2-3 利用範囲

今日、**PostgreSQL** は色々な場面で利用されている。特に **Web** での活用では、**Web** サーバの **Apache**、**HTML** 組込型言語の **PHP** と共に **PostgreSQL** を利用することで容易に **Web** サービスを開発することができる。このためインターネット上では、ショッピングサイトや電子掲示板システムのバックエンドとしての利用されている。また、オープンソースの特徴を生かして、商用製品に組み込んで使われるケースもある。

他分野での活用方法としては、研究・教育分野などでも利用されている。これは多

様なプログラミングインターフェースを持ち、無償で利用できることや、ソースコードが公開されているためである。

4-3-2-4 普及要因

PostgreSQL がこのように普及している要因として、第一にこのような高性能の商用 **RDBMS** は高価で簡単には利用できなかった点が挙げられる。従来、実用的な **RDBMS** が無償で利用できることは想像もできなかったことである。

次に多くのオペレーティングシステムで動作するよう、ユーザーの手によって移植が積極的に進められた点である。これはソースコードが自由に入手できるというオープンソースの特質が生かされた結果である。このため、多くのシステムで稼働する環境が作られた。

また **PostgreSQL** のライセンスは **BSD** スタイルである。このことはソフトウェア開発会社が自社製品に **PostgreSQL** のシステムを組み込んで利用することを可能にしている。これは自社の商用ソフトのソースコードを公開することなく組み込むことが出来るからである。

実際に、**PostgreSQL** を組み込んだシステムの販売や、導入サポートを行う企業が現れている。たとえば日本のシステム会社である株式会社 **SRA**(<http://www.sra.co.jp>) は **PostgreSQL** の開発者を抱えており、また **PostgreSQL** 開発メンバーとコンサルタント契約を結ぶことで **PostgreSQL** のサポート事業を行っている。

4-3-3 PostgreSQL の歴史

4-3-3-1 INGRES

PostgreSQL の原型はカリフォルニア大学バークレイ校コンピュータサイエンス学科で **Michael Stonebraker** 教授らが行ったデータベース研究プロジェクトが元になっている。**PostgreSQL** の原型は、1975年から1977年にかけてこのプロジェクトによって実装された **INGRES** である。**INGRES** は **RDBMS** としては最初期のソフトウェアの一つである。**RDB(Relational Database)** とは、1970年 **IBM** のエドガー・F・コッド博士によって提唱された関係データモデルである。

INGRES はその後抽象型データ形式のサポート、問い合わせ言語である **QUEL** などの機能の拡張が行われた。そして1980年に **Relational Technology Inc.** 社が **INGRES** のサポートと開発を受け継いだ。現在は **Computer Associates Inc.** 社の **IngresII** という製品になっている。

4-3-3-2 POSTGRES

1980年代中頃、Stonebreaker教授らはINGRESの成果を元にPOSTGRES("POST inGRESS"の意)という新たなDBMSを開発した。このPOSTGRESの特徴は従来のリレーショナルデータベースにオブジェクト的概念の機能を搭載したことである。

これは通常のRDBが2次元の表として文字列型や数値型などのデータを格納しているのに対して、POSTGRESは図形や地理情報の型などを扱えるようになるなど、オブジェクト型を使えるようにしたことである。

商用化の動きとしては、POSTGRESの成果を元にIllustra Information technologies, Inc社がILLUSTRAという商用ORDBMSを開発した。ILLUSTRAはその後Infomix Software Inc.社のInfomix Universal Serverのオプションの一つとなり、現在はIBMの製品の一部になっている。

4-3-3-3 Postgres95

POSTGRESは90年代中頃にはプロジェクトとしての開発は終了した。その後はメーリングリストなどのボランティアベースのサポートが行われていた。

1994年カリフォルニア大学バークレイ校の学生であった、Andrew YuとJolly ChenらがPOSTGRESを大幅に書き直した。主な変更点は軽量化と、これまでPOSTGRESで使われていた独自の問い合わせ言語であるQUELから一般的な問い合わせ言語であるSQLに変更したことである。そして1995年にPostgres95(PG-Light名前の出典をチェック)としてリリースされた。Postgres95はライセンス形態がBSDスタイルのライセンス形式になり、商用利用など多くの目的で利用できるようになった。

またPostgres95を元にカリフォルニア大学バークレイ校ではMariposaと呼ばれる分散型データベースシステムの研究が行われた。MariposaのRDBMSにPostgres95のエンジンが採用されていた。Mariposaの成果はその後、Cohera CorporationのE-Catalog Systemとして商用化されている。

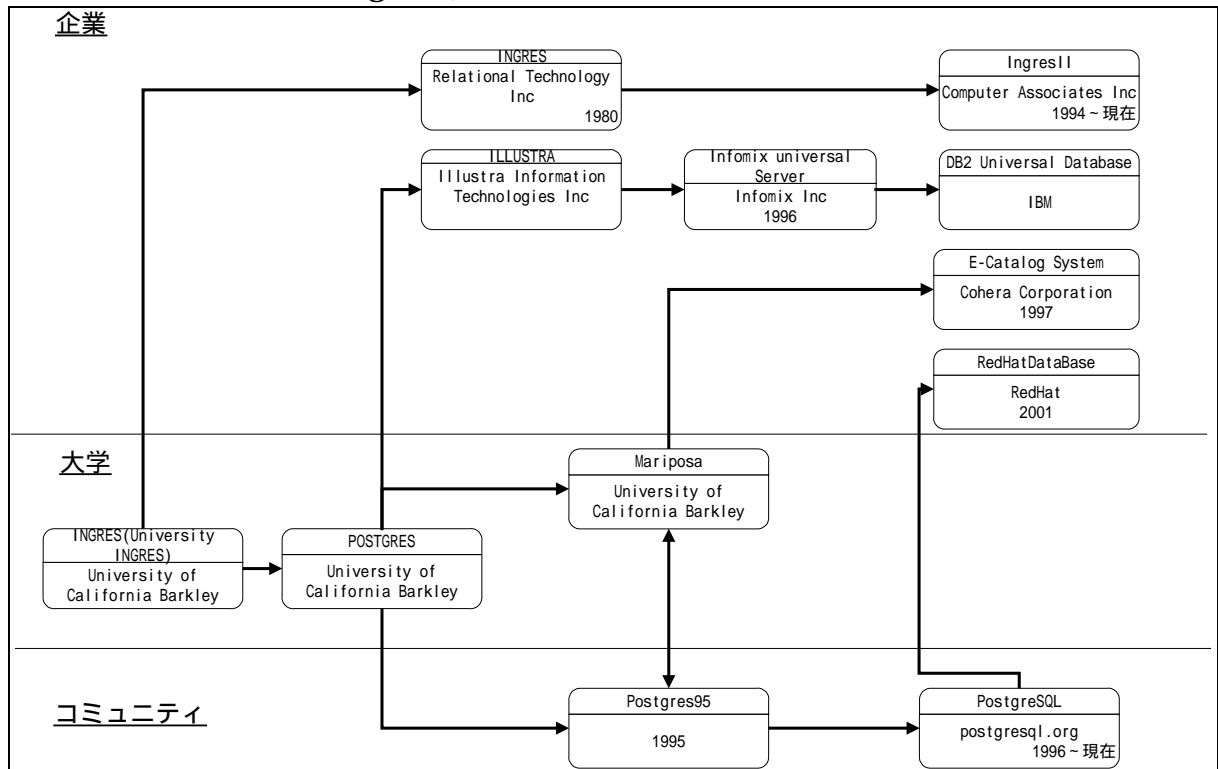
4-3-3-4 PostgreSQL

Postgres95を開発したAndrew YuとJolly Chenらが卒業後、開発は大学からBruce Mojianら6人のコア開発メンバーらのコミュニティに引き継がれた。そして1997年1月にリリースされたバージョンから名称がPostgres95からPostgreSQL(PostgreSQL v6.0)と変更された。その後、安定性と性能の向上が続けられて今日の普及に至っている。2001年12月現在のバージョンはPostgreSQLv7.1.3であり商用RDBMSに匹敵する機能を持ったOSSになっている。

PostgreSQLからは多くのサポート会社や、PostgreSQLを元に商品化されたソフトウェアが誕生している。2001年にはPostgreSQL7.1を元にLinuxディストリビューターであるRed Hat社がRed Hat DataBaseを開発している。

4-3-4 PostgreSQL 発展の流れ

以上の調査を元に PostgreSQL 発展の流れを現わすと次の図のようになる。



【図 4-3 PostgreSQL 系統図】

4-3-5 分析

4-3-5-1 大学がトリガーとなっている

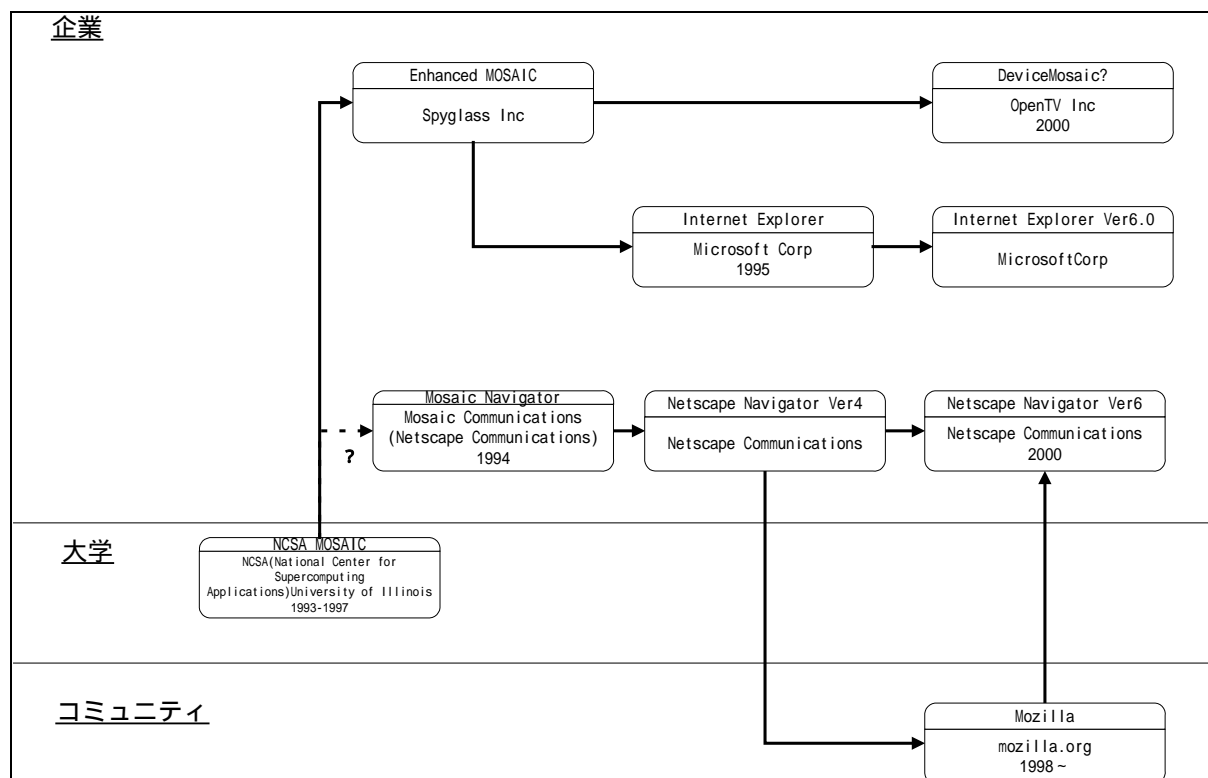
ここでは大学における研究成果がトリガーとなって OSS に発展した事例として PostgreSQL を取り上げた。この事例では、大学のプロジェクトとして誕生した研究成果がオープン化されたことがトリガーとなり、その成果物であるソフトウェアが企業やコミュニティに広く波及していった過程を明らかにした。

このように大学での研究成果が OSS のトリガーとなったケースは PostgreSQL だけではなく、多くの OSS が大学の成果をトリガーにして誕生している。

ソフトウェア名	内容	研究元
XFree86	Xワークステーション用基本フォント、プログラム、文書	マサチューセッツ工科大学
apache	インターネット上で最も一般的に使用されている Web サーバー	イリノイ大学
wu-ftpd	ワシントン大学提供の FTP デモン	ワシントン大学
postgresql	PostgreSQL のクライアントプログラムとライブラリ	カリフォルニア大学バークレイ校
Free BSD	BSD系 UNIX オペレーティングシステム (OS)	カリフォルニア大学バークレイ校
Internet Explorer	マイクロソフトの Web ブラウザ (商用)	イリノイ大学

[表 4-4 大学発 OSS 一覧]

例えばイリノイ大学の学生が開発した **Mosaic** も **PostgreSQL** の事例と同じく、下記の図のようにさまざまに経緯を経て、マイクロソフトのインターネットエクスプローラやネットスケープ・ナビゲーターに発展していることが分かる。



[図 4-4 Mosaic 系統図]

OSS 開発コミュニティを成功させる秘訣としレイモンド(1999)は次のように述べている。

コミュニティ形成を始めるときには、まずなによりも実現できそうな見込みを示せなきゃならない。別にそのソフトは特によく書けてなくてもいい。雑で、バグだらけで、不完全で、ドキュメント皆無でもいい。でも絶対不可欠なのが、開発者候補たちにそれが目に見える将来にはなにか本当に使える代物に発展させられると説得できることだ。(p.94)

OSS コミュニティが形成されるためには、開発に参加する技術者の存在が不可欠である。そのためには、開発技術者を引きつける何かが必要である。それは魅力的な新技術であり、便利な機能を持つソフトウェアである。このような核となるモノが、**OSS** コミュニティが形成されるトリガーとして必要である。

大学で開発されたソフトウェアは、**OSS** 開発コミュニティを誕生させるトリガーになりうる条件を十分に満たしている。

大学での研究成果を元にしたソフトウェアは、新技術としてそのソフトウェアに実装されている。次に個人ではなく大学という組織によって開発されているので、個人では達成することが困難な大規模なソフトウェアを開発できる。そして大学はコンピュータ技術に優れた人材を多く持つ組織の一つである。そのことはより高度で画期的なソフトウェアを開発できる環境であることを意味している。さらに大学は研究成果を発表することで、そのソフトウェアの存在を広く世間に知らしめることが可能である。

4-3-5-2 新たな企業を生み出している

PostgreSQL の事例では、大学の研究成果からオープンソース開発コミュニティが誕生しただけではなく、いくつかの新しい企業が誕生していた。

INGRES や **ILLUSTRA** の事例は、大学の研究成果であるソフトウェアを商用化することで新たな企業が誕生した例である。また大学の研究成果を元に製品化するのではなく、オープンソース開発コミュニティが開発した成果を企業に取り込み、新たな製品として事業化しているケースも存在した。そして **OSS** コミュニティが開発を続けている **OSS** のサポートや導入コンサルタントを行う企業も誕生していることが明らかになった。

これまで日本の大学から企業への研究成果の移転は、学会発表や論文による公開や人材教育を通じた知識の移転によって行われていた。

しかし近年、大学の研究成果の特許化や産学連携による新産業創出などの役割を期

待されるようになった。そして **1998** 年に大学等技術移転促進法が施行された。これは大学の成果を特許化することで企業に技術移転を促進させることを目的の一つにしている。この流れによって **TLO(Technology Licensing Office、技術移転組織)**が各地の大学で設置されている。**TLO** では大学の研究成果を特許化することによってベンチャー企業の育成や新産業の創出が強く求められている。

だが本来、広く社会に還元すべき大学の研究成果が、企業との秘密保持や特許によって行われることは議論の余地がある。

それに対して、**OSS** の事例からは、研究成果のオープン化することで大学の成果を社会に成果を還元することが可能なことを示している。この方法は、大学が研究成果を広く社会に公開する役割を持っていたことを考えると大学と非常に親和性の高い方法だといえる。

さらに特許化によるベンチャー企業育成の場合は、その特許を武器に起業することになる。これでは大学の研究成果から企業が誕生するのは特定の企業に限定されてしまうのではないだろうか。それに対して **OSS** 開発の事例は、直接的に大学の研究成果から企業が誕生する以外の方法でも企業が誕生している。例えば大学の成果が **OSS** コミュニティに渡り、その成果を元に新たな企業が誕生している。また同じ研究成果から複数の企業が生まれている。そのほかにもサポートを行う企業など、多様な企業が生まれていることがわかる。

このように大学の研究成果をオープンソースとして公開することは、これまでにない技術の拡がりを持ち、その結果、多様な形態の企業の源泉となっている。そしてこのような成果が生み出せるのは、オープンソース化が、技術と同時にソフトウェアという”製品”も移転しているからである。

4-3-5-3 新たなコミュニティを生み出している

PostgreSQL の事例では、大学の研究プロジェクトが開発したデータベースソフト **POSTGRES** が **Postgres95** というオープンソースとして発展し、それが今日の **PostgreSQL** につながっている。また **INGRES** や **POSTGRES** も開発完了後もボランティアの手でサポートが行われていた。このように多くの人間が自発的に関わることによって、ソフトウェアが大きく発展した。

一般的な大学と企業の役割は、大学が基礎的研究を行って企業が開発・製造する流れである。つまり大学で生まれた技術は、企業に移転されることによって活用されてきたといえる。そして消費者はこれら企業が生産する製品を利用することで成果を享受することができた。

しかし、大学におけるソフトウェア分野の研究成果のオープンソース化は、成果を活用すべき組織が企業だけではなく非営利のコミュニティなど多様な集団にも拡が

ることを可能にした。このような出来事はクローズされた環境では起こり得ない。オープンにしたからこそ多くの人に関心を持つことができるからである。そして成果を自由に使えるからこそ、関心のある人々が自発的に参加してコミュニティが形成されるのである。

このように研究成果のオープンソース化は、ユーザーでもあり開発者でもある非営利のコミュニティを生みだし、研究成果を広く社会に公開することができる。これは生産者であり消費者でもある集団の出現を予測したトフラーのいうプロシューマーの具現化ともいえるのではないだろうか。

4-3-5-4 三者の密接な関係によって開発速度が促進される

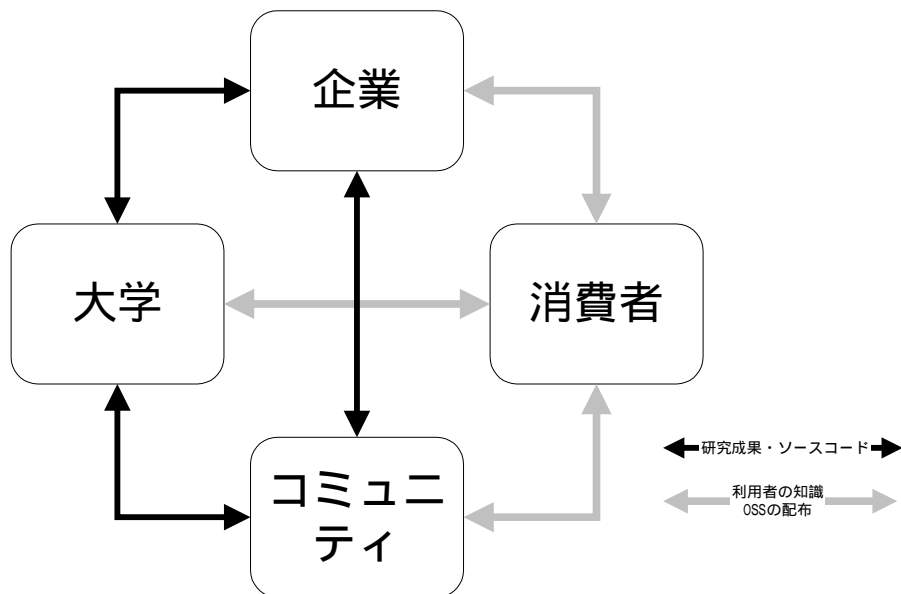
本事例から大学の研究成果を **OSS** 化することによって、企業やコミュニティが新たな形で次々に誕生していったことが明らかになった。そして大学・企業・コミュニティが関係を持ちながらその成果を急速に発展させていくことが分かった。

大学が教育・研究をおこなう役割を担っているのに対して、企業は開発・製造を担っているのは一般的な概念である。そして企業が生産した商品を消費者が消費する。



[図 4-5 従来の研究成果の移転プロセス]

しかし本事例では、大学で研究された成果が企業だけではなく **OSS** コミュニティに成果が移転する事例を明らかにした。ここでは企業ではなく **OSS** コミュニティが開発・製造を担っている。このように本事例ではそれぞれの集団において開発・製造を行っている実体を明らかにした。さらに互いにその成果を利用しあうことで **OSS** がより高機能で信頼性の高いソフトウェアに、急速に進化していったことがわかる。特に開発コミュニティの存在は **OSS** のメリットをフルに生かしており、その結果が今日の普及に至っているのではないだろうか。



[図 4-6 OSS による研究成果の移転プロセス]

このように大学の研究成果をオープンにすることにより、その成果をトリガーに企業やコミュニティが次々に誕生する。これらの組織は、互いに関係を持ちながら研究・開発・利用を行っている。そしてそこから生じた成果を **OSS** として公開することによって、技術が全体にフィードバックされる。そしてその **3** 者の関係は決して固定されたものではなく、**OSS** を中心に自発的に発生した関係である。このような動的なネットワークが研究成果の **OSS** 化によって生まれているのである。

第5章 結論

5-1 大学発 OSS におけるイノベーションプロセス の新規性

これまで大学での技術的研究成果は、企業に技術が移転することによって活用されてきた。大学で発明された技術の特許にして特定の企業に活用させる方法や、企業との共同研究によって開発された技術をその企業が製品化を行う形で、大学の研究成果が社会に還元されてきた。近年、大学の研究成果を元にしたベンチャー企業の育成が議論されている。これも大学の研究成果を元に、新規企業の起業を促進させることで方策のひとつになっている。

これらの方法はいずれも大学は研究を行い企業が開発・製造を行うことが前提となっている。この場合、大学の研究成果の移転プロセスは直線的である。

これに対して大学の研究成果として開発されたソフトウェアをオープンソース化した場合は従来とは大きく異なるプロセスが生まれいている。研究成果の OSS 化は、研究成果が企業だけではなくオープンソース・コミュニティという新しい集団にも移転されている。さらに誰でも自由に成果にアクセスできるため、企業が開発した成果をオープンソース・コミュニティが活用し、オープンソース・コミュニティが生み出した技術を企業が活用するという新しいプロセスが誕生している。また利用者也企業やオープンソース・コミュニティに意見やアイデアをフィードバックさせていることで、開発速度を促進させている。

このように大学・企業・コミュニティ・利用者などが互いに関わり合い、知識や技術をフィードバックすることにより、OSS が進化するプロセスが誕生している。

また OSS の開発者のうち、大学関係者が多くの比率を占めていることがわかった。このことは日本においても大学の研究成果をオープンソース化するうえで、人材については問題がないことを示しているのではないだろうか。このように OSS は、ソフトウェア分野における大学の成果を広く還元する方法として有効に作用しうることが明らかにできた。

5-2 大学における OSS 活用に関する提言

本研究で主張したい点は、ソフトウェア分野における大学の研究成果をオープンソース化して公開することは、新たな大学の知識の還元方法として有効であるという点である。残念ながら日本では、オープンソースに対する理解が進んでいないのが現状である。もちろん日本でもオープンソース化して利用されている例はある。また日本

にも多くのオープンソース・コミュニティが存在しているが、海外の **OSS** に対するローカライズや利用方法を研究するコミュニティがほとんどである。しかしインターネット上の基盤を築くソフトウェアはまだ誕生していない。つまりオープンソース運動は盛んではあるが、本研究で注目していたオープンソースを生み出すトリガーが日本では現れていないのである。その原因の解明は今後の研究課題としたい。

しかしながらこのような経済状況の中、イノベーションを生み出す環境を創り出すことは急務である。とくにソフトウェアは日本が弱いと言われている分野のひとつである。オープンソースの活用がその解決策の一つになりうるのではないかと考えている。そこで最後に大学の研究成果をオープンソースとして生かすためにはどのようにすべきかいくつか提言を行う。

5-2-1 デジタル財の性質を活用する

ソフトウェアをオープンソースとして公開することは、技術の移転と同時にソフトウェアという”製品”も移転されることを意味する。さらにソフトウェア分野における研究成果のオープンソース化は、ネットワーク上で多くの人々の知識や労力を蓄積しながら発展することが可能になる。デジタル財の公共財的側面は、商用ソフトではデメリットとなるが、**OSS** はそのメリットを生かす手段となっている。

このようにソフトウェアのデジタル財としての性質を活用することが、これによって素早く研究成果を実用化し社会に還元することができるようになるのではないだろうか。

5-2-2 OSS が育つ環境の整備を

日本のオープンソース運動は他の地域に比べて決して劣るものではなく、むしろ盛んな地域の一つではないかというのが実感である。さらに本論文でも明らかにしたように、オープンソースにかかわる大学関係者の数も多い。しかし、日本の大学での研究成果を元にした **OSS** で、情報時代の社会基盤になっているソフトウェアが見あたらないのも事実である。

そこで **TLO** や **NPO** などの組織が、大学のソフトウェア分野における研究成果のオープンソース化を促進する環境を整備すべきではないだろうか。これらの組織が行うべき役割としては次のような点についてサポートしたらよいのではないだろうか。

- ・ **OSS** に対する啓蒙活動
- ・ **OSS** のライセンス整備
- ・ マニュアルや翻訳の支援
- ・ コンピュータ・ネットワーク資源の提供

- ・人材の育成

これらを行う上で重要なのはあくまでも環境をサポートする点であり、自発性を重視することである。つまり押しつけるのではなく、自発性を促進させて志あるものが活躍できる場を作る努力が育成には必要ではないだろうか。

5-3 解明点

本研究で以下の点について明らかにした。

(1) **OSS** 開発の原動力は、オープンソース・コミュニティだけではなくデジタル財・ネットワーク・ボランタリの **3** 要素が重要である点を指摘した。

(2) **Linux** ディストリビューションを調査することによって、**OSS** 開発メンバーに多くの大学関係者が参加していることを明らかにした。

(3) 大学の研究成果のオープンソース化によって、大学の研究成果が企業だけではなく多様な集団に移転されるケースの存在を紹介した。そして大学・企業・**OSS** コミュニティのそれぞれが成果を移転しあうことによって **OSS** が発展し、研究成果が社会に還元されていくイノベーションプロセスを明らかにした。

参考文献一覧

- **B.Perens,E.Raymond,R.Stallman,Edited by Chris Dibona, Sam Ockman, Mark Stone(1999), "OPENSOURCES: Voices from the Open Source Revolution" O'Reilly & Associates,(倉骨彰訳 『オープンソースソフトウェア』 オライリー・ジャパン 1999)**
- 國領二郎,(1999),『オープン・アーキテクチャ戦略 ―ネットワーク時代の協働モデル―』ダイヤモンド社,
- **E.レイモンドら(1999),「伽藍とバザール」,山田和哉(編著)(1999),『オープンソースワールド』翔泳社,**
- 松岡正剛・渋谷恭子,情報文化研究フォーラム(編)(1986),『情報と文化』NTT出版,
- **Pekka Himanen,(2001), "THE HACKER ETHIC and the Spirit of the Information Age" Random House,(安原和見・山形浩生訳 『リナックスの革命 ―ハッカー倫理とネット社会の精神―』河出書房新社 2001)**
- **Frederic P.Brooks Jr,(1995), "The Mythical man-month: essays on software engineering Anniversary edition" Addison-wesley Publishing Company,(滝沢徹・牧野祐子・富澤昇訳 『人月の神話 ―狼人間を撃つ銀の弾はない―原書発行20周年記念増訂版』ピアソン・エデュケーション 1996)**
- 金子郁容・松岡正剛・下河辺淳,(1998),『ボランティア経済の誕生 ―自発する経済とコミュニティー』実業之日本社,
- 金子郁容,「インターネット時代の組織はなぜできるのか」,『ビジネスレビュー Vol.46 No.2』一橋大学イノベーションセンター
- **Nicholas Negroponte,(1995), "BEING DIGITAL" Random House,(福岡洋一訳 『ビーイング・デジタル ビットの時代』アスキー出版局 1995)**
- 木村泰之・古瀬幸広(2000),「無償デジタル財とボランティア ―インターネット社会におけるボランティア経済の構造―」,『2000年度立教大学大学院社会学研究科論集第7号』
- **Linus torvalds, David Diamond,(2001), "JUST FOR FUN The Story of An Accidental Revolution" Harpercollins ,(風見潤訳・中島洋監修 『それがぼくにはたのしかった ―全世界を巻き込んだリナックス革命の真実―』小学館プロダクション 2001)**
- 根来龍之・木村誠,(1999),『ネットビジネスの経営戦略 ―知識交換とバリューチェーン―』日科技連,

- ・ 今井賢一・金子郁容,(1988),『ネットワーク組織論』岩波書店,
- ・ 奥野正寛・中泉拓也,「情報化とデジタル化・電子化社会」,奥野正寛・池田信夫(編)(2001),『情報化と経済システムの転換』東洋経済新報社,
- ・ 林紘一郎,「情報財」の取引と権利保護」,奥野正寛・池田信夫(編)(2001),『情報化と経済システムの転換』東洋経済新報社,
- ・ 宮垣元・佐々木裕一,金子郁容(監修)(1998),『シェアウェア ―もうひとつの経済システム―』NTT出版,
- ・ 佐々木裕一・北山聡,國領二郎(監修)(2000),『Linux はいかにしてビジネスになったか ―コミュニティ・アライアンス戦略―』NTT出版,
- ・ 今井賢一,今井賢一(編)川村尚也訳(1989),『プロセスとネットワーク ―知識・技術・経済制度』NTT出版,
- ・ 奥野正寛,青木昌彦・奥野(藤原)正寛・岡崎哲二(1999),『市場の役割 国家の役割』東洋経済新報社,
- ・ 金子郁容,(1992),『ボランティア ―もうひとつの情報社会―』岩波新書,
- ・ Neil Randall,(1997),『The soul of the Internet』 Thomson Computer Press,(村井純 監訳・田中りゅう・村井佳世子 共訳 『インターネットヒストリー ―オープンソース革命の起源―』オライリー・ジャパン 1999)
- ・ Carl Shapiro, Hal R.Varian,(1998),”Information Rules” Harvard Business School Press,(千本倅生 監訳・宮本喜一訳 『ネットワーク経済の法則』IDG ジャパン 1999)
- ・ 加藤みどり,(2000),「企業戦略としてのオープンソース ―オープンソースコミュニティの組織論と外部資源を活用した研究開発の発展に関する考察」科学技術庁科学技術政策研究所 Discussion Paper NO.17
- ・ 後藤晃,(2000),『イノベーションと日本経済』岩波新書,
- ・ アルビン・トフラー,徳岡孝夫訳(1982),『第三の波』中央講談社,
- ・ Karl Frenz Fogel,(2000),”Open Source Development with CVS” The Coriolis Group,(でびあぐる監訳・竹内里佳訳 『CVS -バージョン管理システム-』オーム社 2000)
- ・ 野中郁次郎・竹内弘高,梅本勝博訳(1996),『知識創造企業』東洋経済新報社,
- ・ Steven Levy,(1984),”HACKERS Heroes of the Computer Revolution” Anchor Press/Doubleday,(古橋芳恵・松田信子訳 『ハッカーズ』工学社 1987)
- ・ クリフ・ミラー,(1999),『LINUX 革命 ―オープンソース時代のビジネスモデル―』ソフトバンクパブリッシング,
- ・ 西村 めぐみ,(2001),『「もっと知りたい」人のためのパッケージ管理&構築入門』ソシム,

- 石井達夫,(1999),『PC UNIX ユーザのための PostgreSQL 完全攻略ガイド』技術評論社,
- 「月刊研究開発マネジメント 1998.11 月号」アーバンプロデュース,
- 「Software Design2000.6 月号」技術評論社,
- 桑原 潤,(2001),「OPEN DESIGN2001.8 月号」,
- 「Software Design」 2000.11 月号技術評論社,

参考 URL 一覧

[The Open Source Initiative: Home Page](http://www.opensource.org/)

<http://www.opensource.org/>

[Free Software Licensing \(ResearchIndex\)](http://citeseer.nj.nec.com/243892.html)

<http://citeseer.nj.nec.com/243892.html>

[GNU's Not Unix! - GNU プロジェクトとフリーソフトウェア財団 \(FSF\)](http://www.gnu.org/home.ja.html)

<http://www.gnu.org/home.ja.html>

[コピーレフトって何? - GNU プロジェクト - フリーソフトウェア財団\(FSF\)](http://www.gnu.org/copyleft/copyleft.ja.html)

<http://www.gnu.org/copyleft/copyleft.ja.html>

[The 4BSD Copyright](http://www.jp.freebsd.org/www.freebsd.org/ja/copyright/license.html)

<http://www.jp.freebsd.org/www.freebsd.org/ja/copyright/license.html>

[GNU 一般公共使用許諾契約書 - GNU プロジェクト - フリーソフトウェア財団 \(FSF\)](http://www.gnu.org/copyleft/gpl.ja.html)

<http://www.gnu.org/copyleft/gpl.ja.html>

[Netscape Public License FAQ](http://home.netscape.com/ja/mozilla_org/NPL/FAQ.html#3)

http://home.netscape.com/ja/mozilla_org/NPL/FAQ.html#3

[University POSTGRES 4.2](http://db.cs.berkeley.edu/postgres.html)

<http://db.cs.berkeley.edu/postgres.html>

[PostgreSQL](http://www.jp.postgresql.org/)

<http://www.jp.postgresql.org/>

[Mariposa: A New Approach to Distributed Data](http://s2k-ftp.cs.berkeley.edu:8000/mariposa/)

<http://s2k-ftp.cs.berkeley.edu:8000/mariposa/>

[日本 PostgreSQL ユーザー会](http://www.postgresql.jp/)

<http://www.postgresql.jp/>

[NCSA Mosaic Home Page](http://archive.ncsa.uiuc.edu/SDG/Software/Mosaic/NCSAMosaicHome.html)

<http://archive.ncsa.uiuc.edu/SDG/Software/Mosaic/NCSAMosaicHome.html>

[dW: Open source : オープン・ソースの支援](http://www.ibm.co.jp/developerworks/opensource/000623/j_advocating.html)

http://www.ibm.co.jp/developerworks/opensource/000623/j_advocating.html

[developerWorks : Open source](http://www.ibm.co.jp/developerworks/opensource/)

<http://www.ibm.co.jp/developerworks/opensource/>

[OSDN Japan: Open Source Development Network](http://osdn.jp/)

<http://osdn.jp/>

[JPNIC Top Page](http://www.nic.ad.jp/)

<http://www.nic.ad.jp/>

[木村泰之のホームページ](http://www.ir.rikkyo.ac.jp/~kimura/)

<http://www.ir.rikkyo.ac.jp/~kimura/>

[Think GNU -プロジェクト GNU 日記とソフトウェアの憂鬱](http://www.villagecenter.co.jp/book/think_gnu.html)

http://www.villagecenter.co.jp/book/think_gnu.html

謝 辞

本論文を作成するにあたり、多くの方々からのご支援を賜りましたことを深く感謝致したく、お礼を申し上げたいと思います。

特に本研究において指導教官である亀岡先生には、研究に対する発想力の重要性和心構えについて深く示唆を与えていただいたことに感謝いたします。

また、同じ講座である永田先生には、本研究を進めるにあたって貴重な助言を頂きましたことを感謝いたします。

それから本研究のテーマであるオープンソースに関わる全ての方々に深く感謝いたします。初めてネットワークに触れた日から **10** 年以上の月日が経ちますが、その間、さまざまな出来事や成果を目撃し実感してきました。本論文を書くことが、微力ながらもネットワークの世界への恩返しとなるならば、これに勝る喜びはありません。

最後に知識科学研究科の先生方、亀岡研究室のメンバー、そして **JAIST** で切磋琢磨した全ての友人たちへお礼を申し上げます。本論文は皆様方の力添えがあったからこそ完成できたことを忘れません。本当に有難うございました。