

15 オープンソース・ソフトウェア・ライセンスにおけるソフトウェア特許と商標の扱いに関する研究^(*)

特別研究員 八田真行

Linuxを始めとしたオープンソース・ソフトウェアは、オープンソース・ライセンスの下で自由に利用や改変、再配布が可能とされ、日本でも浸透しつつある。ソフトウェア・ライセンスは基本的に著作権を根拠としたものだが、近年では特許報復条項や商標に関する条項を盛り込み、著作権をてことして著作権以外の産業財産権もコントロールしようとする動きが活発になってきた。加えて最近では、従来、ソフトウェア特許自体を忌避する傾向が強かったオープンソース・コミュニティにおいても、オープンソース向けパテント・プールの形成やオープンなパテント・ライセンスの模索など、戦略的に特許を活用することによって自由な開発を明確に保障しようとする動きも出てきている。この研究では、ソフトウェア特許に関する最新の研究動向や専門家へのヒアリングを踏まえた上で、オープンソース・ライセンスごとに異なる特許や商標に関する条項の要求内容を精査して比較するとともに、オープンソースにおける著作権以外の扱いや様々な事例に関して総合的に整理を行い、今後のソフトウェア開発にどのような影響を与え得るのかを検討した。

I. はじめに

ソフトウェアは一般的にプログラム(アルゴリズム)とデータから構成されている。プログラムもデータも、法律的に創作性のある著作物と認められれば、著作権で保護される。このことは、現在では当たり前となっているが、ソフトウェア、特にその利用については、元々経済的な価値が余り見いだされていなかった。1970年代にはソフトウェアというのは何の価値もないものであると思われていた。それが、80年代に入って、価値が見いだされ、どのような制度で保護するのが適切なかという議論が行われ、著作権法で著作権として保護するようになった経緯がある。

ところが、この20年ぐらいで、ソフトウェアに限らず、著作物は非常に経済的価値があるということが分かってきた。特にソフトウェアにおいては、著作権の保護に加えて、特許、商標など、その性格が全く異なる権利で保護され得る特異な存在であることを考慮しなければならない。もっとも、ソフトウェアそのものが特許で保護されるべきかということについては、代表的事例として米国最高裁判所のダイヤモンド対ディア判決¹などがあるが、欧州などでもまだ議論が続いている。ソフトウェアについては、非常に根強い反ソフトウェア特許運動というのがあり、それらの運動の根底にあるものを理解しないとこの問題はなかなか理解できない。欧州においては、欧州委員会のソフトウェア特許指令案²の是非をめぐって、アンチパテントとプロパテントの両派が非常に激しく議論し、2005年7月6日に欧州議会において大差で否決されたという経緯がある。

そのような状況の中で、オープンソース・ソフトウェアというのは、元々ライセンスされることを前提に開発されているソフトウェアであることから、一般のソフトウェアの権利保護とは異

なるところも多く存在している。

そこで、オープンソース・ソフトウェアにおける、著作権以外に、特許、商標はどのように扱われているかを整理し、ライセンスの在り方についても明らかにするというのが、この研究の狙いである。

II. オープンソースとは

Open Source Initiative³が作成した、「オープンソースの定義(The Open Source Definition)」⁴が規定する、一連の要件に準拠したソフトウェア・ライセンスの下で公開されたソフトウェアを、「オープンソース・ソフトウェア(Open Source Software)」と呼ぶ。オープンソース・ソフトウェアは、原則として誰でも自由に改変したり再頒布したりすることが可能である。

近年目覚ましい普及を遂げたオペレーティング・システムGNU/Linuxを筆頭に、オープンソース・ソフトウェアは我々の生活へ急速に浸透しつつある。多くの家電、携帯電話などには、今や何らかのオープンソース・ソフトウェアが使われており、我々は知らぬ間にライセンシーとなっているのである。

「オープンソース」という用語そのものが登場したのは1998年のことであるとされているが⁵、ソフトウェアを「オープンな」著作権ライセンスによって法的に誰でも自由に利用できる状態に置くという発想そのものの由来は、1980年代のフリーソフトウェア運動にさかのぼる。

オープンソース・ソフトウェアと呼ばれるソフトウェアの種類はいろいろあるが、重要なのは、ソフトウェア・ライセンスにおいて、ソフトウェアが誰でも「自由に」利用できる形態になっているということである。「自由に」利用できないソフトウェアはプロプライエタリな(proprietary)ソフトウェアと呼ばれている。

(*) これは特許庁委託平成21年度産業財産権研究推進事業(平成21~23年度)報告書の要約である。

例えば、GNUプロジェクトの創設者であり、フリーソフトウェア財団(Free Software Foundation)の代表者でもあるRichard Stallmanによるフリーソフトウェアの定義なども広く使われているが、簡単に言えば、オープンソース・ソフトウェアは、多くの場合、無料で誰もが自由に使えるソフトウェアである。そのようなソフトウェアの代表例としては、Linuxとか、ウェブ・ブラウザのMozilla Firefoxとか、最近では、携帯電話のOSなどにも使われているGoogleのAndroidなどが存在している。

この研究を通じて、オープンソースという表現が二つの異なる意味で使われているような印象を持った。上述したような法的側面と並んで、開発形態としてのオープンソースに注目が集まっているという印象がある。具体的には、一般に、バザール型開発といわれるものであり、ヒエラルキーや指令系統のようなものがなく、多くの人数が一つソフトウェアを誰もが自由に開発できる。このような開発形態について、オープンソースという表現が使われる場合も多いと思われる。法的状態としてのオープンソースは、開発形態としてのオープンソースを支え、開発を非常に円滑にしていると考えられる。この研究で、オープンソース・ソフトウェア・ライセンスを扱っているのは、それがバザール型開発にどのような影響を与えているかという見地からである。

開発形態としてのオープンソースに類似した発想をソフトウェア開発以外にも応用する動きが出てきている。有名なものとしては、Lawrence Lessig教授が提唱した、書籍やテキストなどの著作物を誰でも使えるような条件の下で公開し、それによって、著作物の再利用や二次利用、イノベーションの推進等を行おうというCreative Commons⁶がある。

さらに、最近では、オープン・ハードウェアという考え方も非常に注目されている。例えば「Open Source Hardware (OSHW) Statement of Principles and Definition v1.0.」⁷等ができつつある。

また、オープンソースの原則のようなものに影響を受けた活動として、各国の特許庁が行っている発明の特許性審査を誰でも参加できる形態で行おうという、Peer-To-Patent⁸というコミュニティ活動がある。

Ⅲ. オープンソースに関連する先行研究

オープンソースという考え方は経営学にも影響を与えている。Linux、GNU/Linuxが非常に成功を収めたため、そのようなオープンソースの方法論はうまくいく、少なくともある種の条件がそろえばうまくいくであろうという議論が広まった。そうすると、それから何か知見が得られるのではないかという動きが出てきたのは当然のことである。

それらのうち、Chesbrough(2006)は、開発形態としてオープンソース、法的状態としてもオープンな、オープン・イノベ

ーションはイノベーションの促進に非常に有益であり、それに立脚したビジネス・モデルを考えていくべきだということを述べている。これは影響力が非常に強い考え方となり、それ以来、オープン・イノベーション、又はオープン・ビジネス・モデルなどの考え方が出てきた。

Katz & Allen (1982)が論じているNot-Invented-Here (NIH) 症候群という考え方があるが、その打破ということも主要な研究テーマである。すなわちオープン・イノベーションの場合、開発形態としてのオープンソースを考えると、開発人材を含む開発資源等は、外部にもあるので、組織外部との協働を可能にするためにはどうしたらよいかということを考える必要がある。一つの在り方としては、ライセンスを工夫して、外部との協働をスムーズに行うというようなやり方を考えるべきではないかという議論が経営学の分野でも出てきている。これが、NIHのような研究と組み合わせると、オープン・ライセンス戦略とか、標準化戦略という議論において、いろいろ影響が及んでいるところである。

Ⅳ. オープンソース・ソフトウェア・ライセンスの特徴と分類

オープンソース・ソフトウェアは著作権で保護され得るので、第三者が当該ソフトウェアを使用する場合には一般的に著作権ライセンスが設定されることとなる。著作権者は、多くの場合はソフトウェアの制作者であるが、一定のライセンス条件の下で再利用するための許可を出す。代表的な、よく使用されているオープン・ソフトウェア・ライセンス契約書の雛形の幾つかの中から最適な契約書の一つを選んで適用することが慣行として一般的に広く行われている。

オープンソース・ソフトウェアのライセンスの形態は幾つかあるが、多くの場合は、「オープンソースの定義」に規定されている条件を満たしている又は違反していないと考えられるソフトウェア・ライセンスであるといわれることが多い。これにも議論があるところではあろうが、オープンソース・ソフトウェア・ライセンスの形態の非常に有名な例としては、GNU GPL (General Public License)⁹がある。これは、Linux カーネル(Kernel)に適用されているライセンスである。その他には、BSD (Berkeley Software Distribution) License、Apache Licenseなどが存在し、これらのライセンスが実際に使用されている普及の割合は、GPLが5、6割で、BSD Licenseが1、2割、Apache Licenseが1、2割といったところであろう。最近ではApache Licenseが非常に伸びている。

これらのオープンソース・ソフトウェア・ライセンスは、ソフトウェア業界、実際の開発現場でも注目されており、法律的にも注目されている。オープンソース・ソフトウェア・ライセンスについては、2000年代に入ってから、様々な分野で研究が進

んでいる。先行研究をレビューした結果、ソフトウェアが著作物として保護可能であることから、当然ながら、従来の学術研究、又は法曹実務家の関心は、主に著作権ライセンスに注がれていたといえることができる。中でも従来非常に注目されていたのは、コピーレフト(Copy left)性である。コピーレフト性とは、Copyrightの逆という意味の造語であり、互惠性(reciprocity)があるという意味であり、幾つかのオープン・ソフトウェア・ライセンスにおいて、ライセンスされている当該ソフトウェアに何らかの改変を加えた場合、改変を加えたライセンスはそのソースコードを開示しなければならないというような条項が設けられていることを意味し、これは一般にGNU GPLのライセンスの特徴的なものだと思われるが、実は似たような条項は他のライセンスにも含まれている。これに注目した代表的な研究としては、Lerner & Tirole (2002) の論文などがある。

V. オープンソース・ソフトウェアとソフトウェア特許

1. オープンソース・ソフトウェア・ライセンスにおけるソフトウェア特許の扱い

オープンソース・ソフトウェア・ライセンスとは、誰でも自由にソフトウェアを利用できるということを、著作権を根拠として法的に保証しようとする試みに他ならない。一方、ソフトウェアに限らず、特許とは、発明者に排他的独占権を付与し、いわば誰でも自由に利用「させない」ことによって、発明者に利益を還元する法的な仕組みを用意し、発明公開のインセンティブを高めようとする施策である。すなわち、オープンソースと特許は本質的に相容れない部分がある。このことは、オープンソース・ソフトウェアが大きな経済的価値を持つようになり、大規模な特許ポートフォリオを持つ営利企業がオープンソース・ソフトウェアの開発に参加するようになると、無視できない問題となってきた。

問題の複雑化に更に拍車を掛けるのは、ソフトウェアの法的保護手段として、著作権と特許権が完全に独立だということである。すなわち、著作権ライセンスとしてはオープンソース・ソフトウェアでありながら、ソフトウェア特許が成立した技術を実装しているため、特許ライセンスを別途取得しない限り、誰でも自由に利用できるとはいえないソフトウェアが存在する。

オープンソースの文脈で、ソフトウェア特許の問題に対処するための現時点で最も一般的な手段は、オープンソース・ソフトウェア・ライセンスに特許の扱いを規定した「特許条項」(Patent Clause)を含めることである。

オープンソース・ソフトウェア・ライセンスの種類は数多くあるが、使用が余り推奨されていないものも多く、比較的多く使

われているライセンスは七つである。この中で、特許関係条項が全く含まれていないのはBSD Licenseだけであり、Apache License、GPL、LGPL (Lesser General Public License)、CDDL (Common Development and Distribution License)、MPL (Mozilla Public License)、Artistic Licenseには特許関係条項が含まれている。特に、GPLは11条、Artistic Licenseについては13条である。

2. 特許条項の有効性

しかし、そもそも特許条項は法的に有効といえるのだろうか。特許条項について直接争われた例は見当たらないが、オープンソース・ライセンスの法的有効性に関して、法廷において正面から争われた例として著名なのが、いわゆるJacobsen v. Katzer訴訟である。

カリフォルニア大学バークレー校Robert Jacobsen教授は、オープンソース・ソフトウェアJMRI (A Java Model Railroad Interface)の主要開発者の一人である。JMRIは、Artistic Licenseの下で公開されていた。

一方、オレゴン州でKAMIND Associates Inc.なる企業を経営していたMatthew Katzerは、JMRIのコードの一部をコピーし、自らの製品に使用したが、その際、Jacobsenがライセンスを許可する条件として提示していた条項に従わなかった。

そのため、Jacobsenは、Katzerを、契約不履行とともに、著作権侵害で仮差止めを求め、米国カリフォルニア北部連邦地裁に提訴した。2007年、連邦地裁は、問題となったライセンスの条項はライセンスの範囲を制限するものではないとして、著作権侵害による仮差止請求を却下した。Jacobsenはこの判決を不服として米国連邦巡回控訴裁判所(CAFC)に控訴した。

CAFCは、オープンソース・ライセンスの法的強制力を認め、Jacobsenのライセンス条件は、当該ソフトウェアの著作権を許可する条件であるとし、当該ソフトウェアの使用者がこの条件に従っていなければ、ライセンスで許可された範囲を外れるとみなされるため、著作権侵害になるとして、Jacobsen側の本件勝訴の可能性と回復不能な損害につき審理するように連邦地裁に差し戻した¹⁰。

地裁の差し戻し判決では、Jacobsenが仮差止めされなければ、回復不可能な損害を被ることを立証していないとして、仮差止請求を再度却下した。

最終的には、この事件は、2010年、和解によって解決されたが、CAFCの判断として、オープンソース・ソフトウェア・ライセンスの法的な有効性が肯定され、特許関係条項も恐らく有効であろうというのが筆者の考えである。

3. 特許条項の二類型

オープンソース・ソフトウェア・ライセンスには特許関連条

項が含まれている場合がある。典型的なものとしては、Apache License 2.0の第3条「特許実施権許諾」であり、この特許関連条項には、大きく分けて二つの規定が存在する。

(1) 自動ライセンス

第3条「特許実施権許諾」第1文は、Apache License 2.0が適用されたソフトウェア又はその一部の著作権者は、ライセンシー(You)にそのソフトウェアの利用(改変や販売を含む)に対して永続的(perpetual)、全世界的(worldwide)、非排他的(non-exclusive)、無料(no-charge)、ロイヤルティフリー(royalty-free)、取消不能(irrevocable)な特許ライセンスを与えるという構成になっている。

(2) 特許報復

特許報復(Patent Retaliation)は、最近では多くのオープンソース・ソフトウェア・ライセンスに導入されているメカニズムである。

第3条「特許実施権許諾」第2文は、ライセンシーが、ライセンスが適用されたソフトウェアやその一部(Work又はContribution)が特許の寄与侵害又は直接侵害を構成すると主張して特許訴訟を提起した場合、Apache License 2.0の下でライセンシーに与えられた当該ソフトウェアに関する特許ライセンスは打ち切られる旨規定している。打ち切りの後は逆にライセンサーからの特許侵害訴訟を受ける可能性が出てくる。同様の条項はGPLv.3などでも採用されている。

4. 特許関連条項に関する議論

特許関連条項に関する議論の一つに黙示的許諾(Implied license)に関する議論がある。これは禁反言(Legal estoppel)の関する議論ともいえる。自動ライセンスの元々の発想は、黙示的許諾である。オープンソース・ソフトウェアである以上、ライセンス相手に明示的に(explicitly)許可を与えているわけではないとしても、ライセンシーには自由な利用を認めているはずである。事実上実施を許諾した事項を、その有効性を後から否定することは許されない。それを特許ライセンスによって阻害するのは、法的な禁反言(estoppel)となる。

GPLv2ライセンスにおいては、上述の自動ライセンスのような明示的な記載はなく、黙示的許諾になっている。これに関して様々な議論があり、元々のソフトウェアが改変されていない場合は、黙示的許諾は有効であるとされている。ライセンサーが誰かにソフトウェアを渡した場合、ライセンサーが渡したソフトウェアがそのままの状態であるときは、後から何か追加的な制約は許されないであろう。ただし、ライセンシーが当該ソフトウェアに何らかの改変を加えた場合にも、黙示的許諾が認められるかという点に関しては非常に議論がある。

オープンソースの定義に準拠しているのがオープンソース・ソフトウェア・ライセンスという定義であるが、オープンソ

ースの定義に準拠しているということは、事実上、全てのライセンスにおいて黙示的許諾が有効であるとみなすべきだという議論もある。BSD Licenseには特許に関して全く触れられていないが、BSD Licenseがオープンソースの定義に準拠しているのであれば、黙示的許諾が与えられているのではないかという議論もある。オープンソースの定義6条、7条において¹¹、利用分野による差別禁止、権利の分散の禁止が規定されており、利用の目的やライセンシーによって権利を制限することは、オープンソースの定義に違反することになり、そのようなオープンソース・ソフトウェア・ライセンスはオープンソースの定義に準拠しているとはいえないのではないかという議論がある。

MXM (MPEG eXtensible Middleware) Licenseがオープンソースの定義に準拠しているか否かということがかつて議論されたことがある。これは、特許の存在する技術標準をオープンソース・ソフトウェアによって実装する場合に、どのようなライセンス形態を用いるかという議論であり、MPLから特許関係条項を削除して、特許ライセンスは別に取得する必要があるようにするというライセンスを作ろうとして議論になったことがある。

その他の論点として、ライセンス間の差異はそれなりにある。例えば、Artistic Licenseの場合、ライセンシーが当該ソフトウェア及び派生ソフトウェアに関して特許訴訟を提起した場合は、ライセンスは終了すると規定されている¹²。この条項は範囲が広い。Apache LicenseやGPLv3では何らかの形で特許クレームの範囲が限定されている。

特許ライセンスによる直接コントロールが、今後の方向性としてあり得る。これを考える上で注目すべき事例として、Googleが、オープンな形で公開した、Webで使用される動画フォーマットWebMのライセンスの方法が挙げられる。H.264というWebにおける動画の別の技術標準があり、これはGoogleの一存で何とかなる部分ではない。MPEG LAが行っているパテント・プールがこれらに関係するいろいろな特許を扱っているのが、本来ならばGoogleはMPEG LAに参加しなければならないが、そうすると、非排他的で無償というオープンソース・ソフトウェアのライセンス条件を満すことができない可能性が出てくるということで、その点を何とか解決するために考えたのが、WebMライセンスではないかと思われる。

このライセンスの非常に特徴的な点は、Googleが、著作権ライセンスと特許ライセンスを完全に分離し、それを一まとめに提供しようとしている点である。Googleは、ソースコードに関してはBSDライセンスを適用し、Googleが開発したJavaScriptV8の実装に関しては、Additional IP Right Grantという形で、特許ライセンスを与えている。第三者がWebMという技術標準を自分で再実装した場合にも、その技術に関して

Specification Licenseという形でライセンスしていくという点は非常に興味深い。

Googleは、WebMを使用するライセンサーがH.264に関してMPEGLAとそれぞれライセンスを結ばなければならないという事態を避けたいということから、結局、YouTubeという非常に普及したサービスが存在していることを前提に、WebMに関して、第三者が特許侵害でライセンサーを訴えたら、Googleが第三者に提供している様々なライセンスを終了するという、ある種の脅しをかけることによって、オープンソース・ソフトウェア・ライセンスを維持しようとしていると考えられる。

VI. オープンソース・ソフトウェア・ライセンスにおける商標

オープンソースという文脈において、商標によるコントロールという発想は比較的新しく導入されたものである。

オープンソースにおける商標のライセンスは、基本的に、特許同様、著作権ライセンス以外のコントロール手段であり、より直接的には、ある種の品質の維持のために使用されている。

例えば、代表的なオープンソース・ソフトウェアであるLinuxカーネルに関して、1994年まで「Linux」やそれに類似した商標はどこでも取得されていなかった。1994年8月15日、ボストン在住の弁護士William R. Della Croce, Jr.なる人物が米国においてコンピュータOSの分野で「LINUX」なる商標を出願し(出願番号74560867)、1995年9月5日にこの商標は登録された(商標番号1916230)¹³。Della Croce, Jr.はLinuxとは全く無縁の人物である。以降、この人物がRed Hat等のLinux関連企業からロイヤルティ(製品売上げの10%)の徴収を図ったため、1996年からLinuxの開発者リーナス・トーヴァルズを始めとする業界関係者が商標無効を求めて訴訟を起こし、1997年にはDella Croce, Jr.からトーヴァルズへ特許の移転が行われることで和解となった。以来、米国、ドイツ、EU、日本における「Linux」関連商標はトーヴァルズが保有しており、Linux振興団体Linux Foundation内に置かれたLinux Trademark Institute(LMI)が実際の管理を行っている。LMIは、トーヴァルズの商標権を脅かさないと適切に商標表示を行うことを条件に、無料、永続的、全世界的なサブライセンスを行うとしている。

Linux商標のケースは、いわば「商標のオープンソース化」ともいえるケースであったが、オープンソース開発において、商標を用いて何らかの制約を設け、影響力を及ぼそうとするケースも存在する。

オープンソースのウェブ・ブラウザとして著名なFirefoxの場合、ソフトウェアのライセンスはMPLというオープンソース・ソフトウェアのライセンス形態をとっているが、開発を統括する

団体Mozilla Foundationは、ディストリビュータ向けMozilla商標ポリシー(Mozilla Trademark Policy for Distribution Partners)を策定している。このポリシーによれば、Mozilla Foundation自身が用意した無改変のオフィシャル・バイナリ(Unaltered Official Binary)を配布しない限り、「Mozilla Firefox」という名称やロゴ、アートワークなどの利用を禁じている。これは、公式にはMozilla Foundation自らによる品質保証によりMozillaのブランド・イメージを維持、向上させることが目的とされているが、一方でMozilla FoundationはFirefoxブラウザのデフォルト検索エンジンをGoogleとすることで、Googleから年間6000万ドルにも及ぶ資金提供を受けており、オープンソースではあるが、ある部分に関しては変えられたくないという狙いがあるといわれている。

この点に関し、LinuxディストリビュータのDebianは、Mozillaの商標ポリシーは受け入れられないとし、Debianの独自の改良が行われていて、Debianが配布するFirefoxは、ロゴやアートワーク、ファイル名に至るまで全ての名称をIceweasel(Firefox「火のキツネ」に対する「氷のイタチ」)へ変更することとなった。

このようなことは商標においては当然のことではあるが、改変は自由だが、Firefoxを名乗ってはいけないということは、改変が自由であるはずのオープンソースにおいて逆に不思議な感じがする。Mozillaの商標ポリシーの機能と限界ともいえる。上記Debianの例は名前を変えれば解決できる問題であるが、最近の商標を含む訴訟になった事例としては、Sun Microsystemsを買収したOracleが、Googleが、携帯端末用プラットフォームのAndroidにJavaが利用されているため、直接複製したと主張している注目すべき事案がある。この事例では、著作権、特許、商標など様々な争点が挙げられているが、注目すべきは、Javaの商標は、OracleがJavaという技術に対して及ぼそうとしているコントロールの一つの手段でもある。Googleは、アンドロイドでJavaを使いたい、アンドロイドにJavaME Java ME(Java Platform, Micro Edition)というOracleの実装を乗せることがいろいろな事情でできないので、Dalvikという別のJavaの実装を作った。Dalvikは、OracleがJavaの処理を使うときの条件としているJCP(Java Community Process)というJavaの標準化団体のチェックを受け入れたくないため、Javaと名乗ることができない。

VII. オープンソースにおけるその他の対策

近年では、ライセンスの外部に何らかの制度や組織を作り、それによってオープンソース・ソフトウェアの開発を守ろうという動きも出てきている。こういった状況の中、協同ライセンスング(Collaborating Licensing)と呼ばれる形態で研究活動を行うという動きが出てきている。共同ライセンスングとは、皆で何

かを研究開発するときに、著作権、特許、商標などを一つ一つ、ライセンスし合っていると、工数が掛るので、これをもう少しうまくやる方法はないかということで、いろいろ検討されている。ここでは、大きく分けて、弱い協同ライセンスと強い協同ライセンスの二つに分類する。

1. パテント・プール

弱い方は例えばパテント・プールである。パテント・プールは、基本的に参加者の自発性が求められるが、参加者が、持っている権利のうち、余り重要でない一部の特許だけをパテント・プールに提供することがあり得る。

パテント・プールは家電やソフトウェアの業界において普及した対策となっているが、オープンソースにおいて導入されたのも比較的最近である。代表的なものとしてPatent Commons Projectがある。2005年11月15日、Open Source Development Labs (OSDL)によって開始されたプロジェクトであり、現在ではOSDLが発展的に解消して創立されたLinux Foundationの下にある。

2. ディフェンシブ・パテント・ライセンス

強い協同ライセンスの具体例としては、DPL (Defensive Patent License)がある。

DPLは、カリフォルニア大学バークレー校(University of California, Berkeley)のJason SchultzとJennifer Urbanが、最近提唱している考え方であり、これは特許ライセンスにGPLの考え方を入れたものである。著作権ライセンスによる間接コントロールというようなことではなく、直接的に特許ライセンスでコントロールしようという動きが出てきたのが、DPLの注目すべきことである。

3. 貢献者協定

既存のオープンソース・ソフトウェアを改変するとき、自分が書いたソースコード等を公開するには大きく分けて二つの方法がある。一つは、自分が書いた部分のみを単に公開することであり¹⁴、もう一つは、現在そのソフトウェアの開発を主導的に担っている主体(開発プロジェクト等)にコードを提供し、彼らが管理するソースコード(メインラインのコードベース)の一部として統合(マージ merge)してもらうことである。例えばLinuxカーネル向けに新たなデバイスドライバを書いたとして、単に自分のウェブサイト等でデバイスドライバのソースコードを公開するのが前者であり、Linuxカーネルメーリングリスト等に連絡し、リーナス・トーヴァルズ以下主要な開発者の同意を得てLinuxカーネルの標準的配布物の一部として受け入れてもらうことを目指すのが後者に当たる¹⁵。

後者の場合、提供を受けた部分の著作権の扱いが問題となることが多い。開発プロジェクト側がコードのマージに当た

って特に条件を指定しなければ、当然提供されたコードの著作権はそのコードを書いた人物に帰属したままとなる。例えばLinuxカーネルはコードのマージに関して(コードの品質が良いという以外に)特に要件を指定していないので、Linuxカーネルは膨大な数の著作者による共同著作物と考えることが出来る。この場合、Linuxカーネルという著作物全体に関わる意思決定(ライセンスの変更等)においては、理論的には著作権者全員の意思確認が必要となる。また、後になって、「真の」著作権者(と名乗る主体)が登場し、自らの同意なく勝手に著作物をマージされてしまったと主張する場合もある。

これに対し、一部の開発プロジェクトではコードの提供に際し著作権譲渡証書(copyright assignment)の提出を要件としている。著作権の譲渡に関しては、近年になって普及しつつあるのが貢献者ライセンス協約(Contributor License Agreements, CLA)である。CLAは、コードを提供しようとする元々の著作権者が、著作権を譲渡する代わりに、提供される側に対してコードの無制限な利用を認めるという形式を取る。

VIII. おわりに

以上、オープンソース・ソフトウェア・ライセンスにおける特許・商標の扱いについて述べてきたが、オープンソース・ソフトウェアについて、なぜここまで特許を制限しようとしているのかという疑問があると思う。この研究において、欧州に行きいろいろな関係者と話をすると、元々オープンソースとも関係なく、根強い反ソフトウェア特許感情というのがエンジニアの間にはある。しかも、反ソフトウェア特許を奉じる人たちが始めたオープンソースは非常にうまくいっているということもあって、彼らも非常に力付けられる部分があって、ソフトウェア特許の存在意義が近年大きく揺らいでいるというのが背景としてあるということが分かってきた。

経済学や経営学において、特許の正当性を主張する研究というのは実はほとんど存在しないということも分かってきた。代表的な研究としては、Boldrin & Levine (2008)、Bessen & Meurer (2008)、これはPatent Failureという標題が付けられており、特許制度自体がだめだという議論である。

この議論について、典型的な例として、よく挙げられるのはジェームズ・ワット(James Watt)の蒸気機関の開発に関するものである。ワットは、特許913号を取得し、資金も有していた。特許の支持者から見れば、蒸気機関の研究開発には、そもそも元手として高額な資金が必要であり、莫大な初期投資を後で回収するためには特許が必要であるという議論である。ただ、逆に、Boldrin & Levine (2008)によれば、ワットは、特許を取得したが、結局、特許で他人から金を得ることに尽力し

てしまい、ワット自身が、特許権が満了するまで蒸気機関を作っていたわけではない。むしろ、ワットの特許は、社会全体から見れば技術発展を妨げただけだったのではないかという議論がある。これは特許一般の話であるが、ソフトウェア特許の有用性を主張する研究というのは更に少ない。

日本の特許庁では、ソフトウェア特許、特に、ビジネス・モデル特許に関しては、開示要件や実施可能要件などについて審査をかなり厳格にしているというようにいわれている。

オープンソース・ソフトウェア・ライセンスにおける特許の扱いについて、基調として常にネガティブな方向になっているのは、元々、反ソフトウェア特許という考えがあって、それがオープンソース・ソフトウェア・ライセンスに影響を与えているからだということは理解するべきである。コピーレフト条項が、著作権をてこに著作権を実質的に無力化するものだったのと同じように、特許権をてこにソフトウェア特許の問題を解決するための工夫としてとらえられるであろう。

誰にも差別せず、無償でライセンスするというのがオープンソースの原則であるので、特許の独占的排他性などのように平仄を合わせていくかというのが難しくなってくる。それを、このような形で何とかつじつまを合わせることができようかというのが、オープンソース・ライセンスにおける特許の扱いであると思われる。

商標に関しても同様に、オープンソース・ライセンスを維持しながら、実質的にコントロールする手段として考えられている。これらの点を理解しないと、今後、ソフトウェアをライセンスするとか、オープンソース・ソフトウェアを特に経済的な利益を求める形で利用したいという場合には、非常に問題になるのではないと思われる。この研究がこういった需要に応えることができるものとする。

license to make, have made, use, offer to sell, sell, import and otherwise transfer the Package with respect to any patent claims licensable by the Copyright Holder that are necessarily infringed by the Package. If you institute patent litigation (including a cross-claim or counterclaim) against any party alleging that the Package constitutes direct or contributory patent infringement, then this Artistic License to you shall terminate on the date that such litigation is filed.

¹³ USPTO Assignments on the Web

(<http://assignments.uspto.gov/assignments/q?db=tm&rno=1916230>) 2011年3月25日閲覧。

¹⁴ この場合、コードはユーザが手元で自ら本体に追加(典型的にはパッチを当ててビルド)することとなる。

¹⁵ Linuxカーネル本体にマージされた日本由来のコードとしては、2009年にマージされたTOMOYO Linux等がある。

¹ DIAMOND v. DIEHR, 450 U.S. 175 (1981).

² The proposed directive on the patentability of computer implemented inventions, Commission proposal COM(2002) 92.

³ Open Source Initiativeについては公式ウェブサイト <http://www.opensource.org/osd.html>参照。

⁴ オープンソースの定義については、Open Source Initiative のウェブページ <http://opensource.org/docs/osd> 参照。

⁵ オープンソース・イニシアティブのウェブサイト <http://opensource.org/history> 参照。

⁶ クリエイティブ・コモンズのウェブサイト <http://creativecommons.org/>参照。

⁷ <http://freedomdefined.org/OSHW>

⁸ 当研究所ウェブページ <http://peertopatent.jp/>参照。

⁹ GNU GPLv3 の原文テキストは、GNU のウェブサイト

<http://www.gnu.org/licenses/gpl-3.0.html>に掲載されている。また、この原文テキストの仮訳は、独立行政法人情報処理推進機構(IPA)のウェブサイト <http://ossipedia.ipa.go.jp/legalinfo/gpl-3.0j.html>に掲載されている。

¹⁰ CAFC の判決文は、<http://www.cafc.uscourts.gov/images/stories/opinions-orders/08-1001.pdf> から入手可能。

¹¹ The Open Source Definition

6. No Discrimination Against Fields of Endeavor

The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.

7. Distribution of License

The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.

¹² Artistic License 2.0

(13) This license includes the non-exclusive, worldwide, free-of-charge patent