Intricacies of Software Protection: A Techno-Legal Review

Shabib Ahmed Shaikh and B R Londhe

1CSIR-URDIP, NCL Campus, Sr.No.113, 114, Pashan, Pune, Maharashtra 411 008, India
2SIMS, Symbiosis International University, Pune, Maharashtra 412 115, India

Received 15 August 2015; accepted 29 February 2016

The software industry is experiencing rapid technological advancements along with heavy R&D investments and tougher competition. It is seeking concrete protection of its work within the available legal structures of IPR. Though the IPR system provides protection for almost all types of creations from human minds, it is proving to be insufficient in protecting software under the different IPR tools available. Each of the IPR tools has advantages, but when considered for protection of software, their limitations weight more than their advantages. An attempt is made in this paper to put forth a review about the complexity of software and its protection in accordance with the available tools for IPR protection.

Keywords: Intellectual property rights, software patents, copyright, trade secret, software protection

Intellectual Property refers to the creation of human minds. Intellectual property rights (IPR) give the creator an exclusive rights over the use of his/her creation for a certain period of time covering products, services, methods of doing business, designs or representations of ideas and arts. IPRs are gaining higher significance day by day as they are not only helping to protect the rights of a researcher but also encourage research and innovation in new fields of technology. The resulting innovation in turn is generating substantial wealth for nations by means of foreign investment. Hence, countries are devising ways to cultivate the knowledge and innovations generated by means of intellectual properties. IPR offers a wide range of protections such as patents, trademarks, copyright, design, trade secrets, anti-competitive practices, protection of new plant varieties and data protection.

Protection of software under IPR is itself debatable. The TRIPS Agreement is ambiguous on protecting software. Even then, there is a worldwide acceleration inclined in favor of adopting patent protection for software. However, countries are at their own discretion to grant protection for software which can be protected under trade secrets, copyrights and patents. Some countries encourage protection of software and have laws to protect them under some IPR protection, while there are many which do not.

Research Work in the Field of Software Protection

Research work in the field of software protecting has been taken up since the 1960’s when the software industry had just started as a separate entity, seeking an identity of its own, especially in the USA. Studies had also put forth the idea of protecting software via copyright, patenting and trade secrets. Further, the licensing deals to commercialize the software under copyrights are simpler and cheaper. However, authors have also debated over protection of computer program as copyright and its use.

Yang, 2012 has highlighted that copyright automatically authorizes ownership and has less fear for infringement. Further, the licensing deals to commercialize the software under copyrights are simpler and cheaper. Watt, 2007 argued that copyright alone either leaves important aspects unprotected, or overprotected. Yang, 2012 has highlighted that copyright automatically authorizes ownership and has less fear for infringement. Further, the licensing deals to commercialize the software under copyrights are simpler and cheaper. However, authors have also debated over protection of computer program as copyright and its use.

Literature related to trade secrets for software protection also existed as an alternative to copyrights since beginning. Raysman (1978), argued that patents and copyrights are generally unrealistic for computer program protection because of the necessity of disclosure. Trade secrets are an effective tool for...
protection of software. Authors have also focused on whether trade secret protection can be used to protect information that is stored in the cloud when an act causes information to flow from the owners storage to a third party storage.\textsuperscript{18}

Studies related to software patenting and business methods have also been carried out. As software and its services are a new subject matter for patents, it is still debatable whether a software should be patented or not and to what extent in different countries. The impact of software patents, their quality on innovation and their effects have also been studied. Studies related to impact of IPR in general and specifically with respect to computer software in developing countries highlights that laws for protecting software are not stable because of the newness and constantly evolving nature of the software technology even though laws in the US\textsuperscript{20,21} and Europe\textsuperscript{22} have changed accordingly over a period of time to adopt to the changing technological advancements in the field of software. Studies have also highlighted that there is less exposure about the rules and procedures for obtaining software and business method patents, when compared to familiarity with patents for other products.\textsuperscript{23} Studies in favor of software patents have also been carried out.\textsuperscript{20,24} In-depth studies exploring the general characteristics of software patenting in the last couple of decades highlighting the rapid growth in software patenting and its effect on R&D were also carried out showing that software patenting does not relatively increase R&D and strategic patenting by non-R&D firms may pressurize firms to engage in a patent "arms race".\textsuperscript{25}

Many studies have been carried against private (patented) software, highlighting the fact that software patents obstruct innovation and open source is a way to foster quick innovative development. Studies have pointed out that the open-source model is an alternative to patented software and a proven commercial model by itself.\textsuperscript{9} However, studies have also pointed out that companies investing in open source are acting in a monopolistic way by driving the open source community to suit their patented applications and software which sit on top of the open source applications.\textsuperscript{27-29} Even companies supporting open source are building strong patent portfolio. For example, Red-Hat's patent portfolio is ranked 50th in the Information Technology Industry by the Patent Board.\textsuperscript{28} Red Hat holds 137 granted U.S. patents as part of its intellectual property portfolio as of 2011. A pilot search using Derwent Innovation Index for patents of Red-Hat showed a count of more than 1400 published patents which were either applied or granted as of 2014. Red-Hat, is making profit from the sales, service and support of Linux even though Linux is open source.\textsuperscript{30,31} Google is also slowly building up a strong patent portfolio. IBM, Oracle and others are also following suit even though are a strong publisher of open source.\textsuperscript{9,27}

Studies comparing the laws and patenting trends in software have also been done. Many studies have highlighted the case of USA since the software market was born in the US and it still acts as a trend setter for software patenting.\textsuperscript{30,25} Some studies have also concentrated to understand and interpret laws of specific countries like Sterckx & Cockbain (2010);\textsuperscript{44} Walaski (2004), in his paper attempts to explain the true position of software patents and what lies ahead for the UK industry from the outcome of the software patent debate.\textsuperscript{44} Goswami & Yadav (2010)\textsuperscript{31}, tried to compare the different policies and laws related to software patents governed in the top eight leading software countries. Nandi & Mohanty (2010), highlight the software laws in US, Europe and India highlighting that although software is protected under copyright law, the emergence of software patenting has provided software companies an alternative protection to protect their product.\textsuperscript{39} Pai (2007) discusses the software patenting activities in the US, Europe and India stating that the US, European and the Indian systems are almost the same. However, the Indian system is incomplete due to dearth of specific relevant case laws / judgements and suggests a few focused solutions that can be helpful to the Indian Software Industry.\textsuperscript{32} The European Patenting Office highlighted that in many cases the innovative part of a new product or process lies solely in a computer program. The report doubts whether innovators would make an effort if they did not expect to benefit economically from their work given the fact that the amount of R&D resources put into the development and commercialization of new products is enormous.\textsuperscript{33}

**Software Protection Intricacies**

Ease of copying of the software is the biggest hurdle for any developed software product.\textsuperscript{34,35} It results in financial losses to the software industry,
hence, creating major disincentives to development of new and innovative software programmes hindering the software development. Developers are, hence, seeking for some legal protection of their work. On the other hand, software has a hybrid nature since it can be divided into many different categories depending on its complexity, applicability and implementability. The software’s hybrid nature has forced a re-examination of its protection. This has also triggered debates from time to time, starting from the 1960’s till date, over the protection of software, as software can be protected under laws of trade secrets, copyrights and patents.5-10

Protecting Software under Trade Secrets

Software can be protected under trade secret. Trade secrets offer great duration and it may therefore continue indefinitely as long as the secret is not revealed to the public. However, trade secrets are easily undermined and invalidated through reverse-engineering, independent invention and inadequate diligence in maintaining that secret.39 Trade secret protection of an invention does not provide the exclusive right to exclude third parties from making commercial use of it. A trade secret is more difficult to enforce than a patent. The level of protection granted to trade secrets varies significantly from country to country, but is generally considered weak, particularly when compared with the protection granted by a patent. A trade secret may be patented by someone else who developed the relevant information by legitimate means.

The objective of trade secret is to keep innovations confidential, preventing competitors from learning those innovations. However, in case of software, it cannot be concealed from the competitors once the product comes in the market, as it can easily be copied and decoded.35 Thus competitors can freely imitate the idea behind the software. Apart from the above, employee mobility between competing firms mitigates the problem of secrecy by allowing others to benefit from the programmer’s general experience.5 Hence, trade secrets are the weakest type of protection in the IPR system for software.

Protecting Software under Copyrights

Copyright is a form of protection provided to the authors of “original works of authorship” including literary, dramatic, musical, artistic, and certain other intellectual works, both published and unpublished. Copyright generally gives the owner the exclusive right – to reproduce the copyrighted work, to prepare derivative works, to distribute copies or phonorecords of the copyrighted work, to perform or to display the copyrighted work publicly. A copyright protects an original work expressed in the tangible and fixed form in which it has been set down. It protects only the expression of the work, and not the idea underlying the work.35 It does not insulate inventive ideas from infringement.40

Copyrights are stronger compared to trade secrets, but in the case of software, it does not provide any real protection as the core concept in the software is not protected by means of copyright. Also, a software can be easily copied or decoded and with minor changes it can be reproduced and distributed without the consent of the original author since programs are easy to disguise and their format inexpensive to alter without affecting their substance.5 Secondly, copyright is granted to an author, and in the case of software, many a times, the code may be written by many virtual programmers sitting in different geographical locations.

Protecting Software under Patents

A patent is a form of intellectual property41 granted by the government in order to secure a legal protection for inventions by means of exclusive right for a limited period in exchange for the public disclosure of an invention.42 Patents are also important for trade and industry worldwide as they attract foreign investment and accelerate technology transfer.43 Patents also promote innovation by disclosing an invention in public domain.44-47

Software patents,48 on the other hand are debatable as they are in their infancy49 and some countries encourage them and have laws to protect them in one way or another, while there are many which do not. More than half of the 170+ countries in the world that grant patents, permit the patenting of software-related inventions, at least to some degree.50

Software patents are used to protect the specific software code and/or routines that allow the software to perform certain unique functions. Patents, unlike trade secrets and copyright, provide protection against reverse-engineering, and prevent others from reproducing it.20,29,31 Hence, patents hold the strongest relative protections for software.

As software are becoming more complex and sophisticated with value added features, companies and developers expect their work to be protected in the same way as any other product or service in the
form of patents. This has led to the emergence of software patenting as a new field for protection. However, with the emergence of software patents, new challenges and problems have cropped up since defining software itself has become difficult, leading to confusion in law surrounding the patentability of software and computer related inventions.\(^{52}\) The key points highlighted below have come to the fore and caused a concern for the software industry worldwide for patenting of its software.

**Software is Not a Single Physical Product**

Software is a set of logical programs and differ from the physical object patents as they do not have physical characteristics and other implementation details.\(^{35}\) Software is defined by the functionality it possesses and is developed as an incremental process. It is a pluralistic product that contains several elements, each of which could fall into different categories of IP laws.\(^{53}\) Thus, software patents are harder to design around and tend to survive a long time. Apart from this, patentable inventions in computer science are not whole software programs but individual ideas or approaches to particular problems.\(^{35}\) These properties of breadth and non-obsolescence also mean that software patents can end up covering after-arising technologies that their inventors did not envision — leading to ambush litigation.\(^{54}\)

**Confusion over the Definition of Granting of Patent by WIPO**

Abstract Ideas cannot be Patented

The definition of software and what can or cannot be patented has made a chaos amongst law makers on the protection of software under the patent regime. As Patents are granted on any invention – having 1) patentable subject matter 2) novelty 3) inventive and non-obviousness 4) applicable to the industry,\(^{55}\) software do not fall under it. Software is a program which is a collection of logical instructions generally called algorithms. Algorithms are mere expressions of abstract ideas and cannot be patented.\(^{56}\) However, when the software is embedded in a machine or a process as its part, can be patented.\(^{14,57,58}\)

The TRIPS Agreement highlights two provisions related to protection of software. Its Article 10(1) suggests computer programs can be protected as literary works,\(^{59}\) and Article 27(1) mandated member countries to provide patent protection for inventions in all fields of technology\(^{60}\) which does not exclude software from patenting. These two articles many a times create confusion for nations while implementing software protection and nations highlight only one of the articles based on the rule of interpretation of statute (harmonious construction). The countries supporting software patents stress for Article 21(1) whereas those supporting copyright for software stress on Article 10(1).

**The Rules for Software Protection is Different in Different Countries**

For each country, the exact nature of software patentability is a complex matter.\(^{61}\) According to Section 3(k) of the Indian Patents Act, a computer program per se is not patentable.\(^{62}\) However, a software can be patented in India under Computer Related Invention (CRI) as a product.\(^{63}\) Same is true for South Africa and New Zealand where computer programs are excluded from patentability and are protected under copyrights.\(^{64}\) In the United States, almost all type of the software can be patentable\(^{65-69}\) whereas in the Europe, its Article 52(2) excludes computer programs (as such) from patentability. However, Computer Implemented Inventions\(^{70}\) (CII) – even in the form of a computer program - can be patentable if the application of the software has a “technical effect”.\(^{71,72}\) Hence, it can be said that software is patentable in Europe under CII in the same way as under CRI in India. Japan supports software patents as software-related inventions.\(^{73}\) Australia, South Korea and Canada support software patenting and business method patenting.\(^{74}\)

The IT industry is at a loss when it comes to protection of its software under different regimes, since rules governing each country for the exact nature of software protection and its patentability is a complicated question.\(^{75}\) This has thus, resulted in the failure of the Patent Cooperation Treaty at least for the software protection since, a single international software patent application does not hold valid in multiple regimes supporting different frame works for software protection.

**Software is a Fast Moving Industry**

IT innovations are usually incremental, fast changing and having a short lifecycle. Since, the software industry is moving with a rapid pace in technology and continuously evolving, the laws are also required to be changed and adopted accordingly in tandem with the technology. However, the national patent issuing offices are not updated with the changes in the technology and hence are a cause of a
concern to the industry. Even the time frame for issuance of a software patent in many cases takes more time than the technology life cycle and by the time a patent is granted its technology is absolute.

**Patent Trolls**

Patent troll is a term for companies and individuals who acquire patents for the purpose of extracting money from product developers. Trolls are only interested in monetizing their patents. This “patent trolling” business model has become popular in recent years and is thriving. It accounts for a majority of all patent assertions, winning both larger judgments and larger settlements than the firms that practice patents, and do so despite complaints and some evidence that they assert weak patents. Also known as “patent assertion entities” (PAEs), or “non performing entities” (NPEs), patent trolls are now in every field of technology lately and are mostly a problem for the software industry. The patent trolls are now days equipped with the latest technological patents and are in pursuit to catch hold of infringers forcing competitors and new entrants to license-in their technology at higher royalties. The patent trolls tend to lose less compared to the operating companies since they never cross license a technology and insists of higher royalties. As pointed out by Bessen & Hunt, strategic patenting by non-R&D firms may pressurize firms to engage in a patent “arms race”. This may in turn cause a problem of patent thickets.

**Patent Thickets in Software**

Patent thickets also known as "patent floods", or "patent clusters" are intentionally dense webs of similar overlapping intellectual property rights in a specific technological domain, owned by companies that can hinder further development by other entities and new entrants. Companies construct a "patent thicket" to prevent competition and curb competition or to force others to license their technology. Patent thickets are also used to defend against competitors designing around a single patent. Patent Thickets in software is a concern as it does not give the freedom to operate to the competitors and new entrants, and force them to license or cross license their technologies. To avoid patent thickets and trivial patenting, Zekos (2006), suggests that the patentability subject matter in case of software must not be its implementation but its end result as a product. However, since, a software is an incremental development process, most of the companies try to patent their software activity even with a minor incremental development thereby leading to patent thickets.

**Software Patent Infringements and Litigations**

As software development is an incremental process and is developed on existing inter-related technologies, technical over-mapping is common. Software products commonly use technologies claimed by thousands of patents. This has led to a number of patent infringements and litigation cases not only by patent trolls but also by practicing entities. The number of patent infringement and litigations has also gone up in the countries that grant patent protection for software. Software patents are nearly ten times as likely to be litigated as other patents and hence are more susceptible to abused by trolls. The number of patent law suits filed in United States each year has almost tripled in the last two decades to 3,260 in 2010 in the software patents. One analysis reported that as of 2012 NPEs accounted for the majority of patent infringement litigation filed in the United States, compared to less than a quarter of patent infringement law suits filed in 2007. The number of patent lawsuits filed spiked by almost 30% in 2012 to over 5,000.

Microsoft sued Motorola; Motorola sued Apple and Research in Motion; Research in Motion sued Visto, a mobile technology company; Google, through its Motorola unit, sued Apple, contending that Siri had infringed on its patents. (Google dropped the suit afterwards, leaving open the possibility of refiling at a later date). All of those companies have also been sued numerous times by trolls. According to a Stanford University analysis, Apple and Google, spent as much as $20 billion on patent litigation and patent purchases in 2012 — an amount equal to eight Mars Rover Missions. This spending exceeded spending on their research and development activity. Patent infringements and litigations is a cause of concern for the industry since the litigation fees are in millions of dollars and so also are the damages.

**Grant of Trivial and Broad Category Software Patents**

The expression "trivial patent" is frequently used for patents which third parties think should have not been granted because they lack novelty or inventive step, which means that the proposed invention is seen as either being known already or being too obvious to a technically skilled person ("person skilled in the art") to qualify for patent protection under the
applicable law. As the patent trolls are becoming more and more sophisticated with technology, software companies are applying for any and all types of patents that can be applied in order to build a safe patent portfolio and avoid litigations for infringement. In many cases, a developed piece of software code is drafted for patent application in such wordings that it simply fits in the criteria of patentability – novel, inventive & non-obviousness and applicable to the industry – even though it may not be technically useful as a single entity. This has resulted in the cases of issuance of trivial patents in the field of software.

Lack of Prior Art in Software
Prior to granting of a patent, the patent examiner checks whether there exists any technology related to the said patent application known as prior art. Examiners knowledge about a patents newness and non-obviousness comes from prior art and published patents. However, software patents do not disclose how the claimed software actually functions. The terminology used in software patents is many a times deceptive and hence almost no prior art is available in the said technology. Even the knowledge of the patent examiner in the said field is limited. This many a times leads to insufficient scrutiny of poorly disclosed software patents. Hence, lack of proper prior-art and knowledge of Patent Examiner in the said field of technology is also a concern leading to trivial patents or patent thickets.

Conclusion
Software is being used in almost every field of technology and hence, has become an integral part of technological innovation. Software companies naturally seek to protect the technologies they develop. Software can be protected using trade secrets, copyrights or patents. Patents were designed for new and useful inventions, whereas copyrights were for original expressive works. Trade secrets offer great duration, and possibly high claims for damages when violated, but are easily undermined and invalidated through reverse-engineering, independent invention and inadequate diligence in maintaining that secret. In the case of software, software can be easily copied or decoded. Copyrights are stronger, but only protect the expression of an idea and not the idea itself. Copyrights not always fairly balance since, the copyright creates rights and responsibilities for both the creator and the users and many a times the law is abused by the end user. Trade secrets and copyrights offer only extremely limited protection for the underlying techniques and ideas embodied in the software. Patents thus hold the strongest relative protections for intellectual property. However, software patents are debatable as they are continuously evolving. New technologies have often presented problems for the patent and copyright systems. Countries patenting software also face problems due to the fast technological advancements in the software field, lack of effective changes in the rules and guidelines of the Patent offices, patent thickets, litigations cases, danger of infringements and different laws in different countries. Even then, software patents are not bad as hyped by the media and others. Prohibiting software patents will be bad for business and it will enhance uncertainty. Firms those support Open Source Software are patenting their work. Research studies have highlighted that motivation for further innovation comes only if there is a strong belief for the protection regime and industry will carry out research and innovation only if there are stronger IPR laws to protect their work. Patent protection thus plays a key role in innovation strategies and many important innovations have reached the marketplace with the help of the patent system. It is therefore, felt to have a different framework for software patents where these software patents could be granted within months of their filings and their protection tenure be for a shorter duration of time. Software can then be protected under different tools of IPR. During the initial phase of software development, the protection can be applied of trade secrets. After development of the software, the code and the GUI can be protected under copyrights and the software itself can be protected under the frame work of patents.

References
2 http://www.wto.org/english/tratop_e/trips_e/intel1_e.htm (accessed on 7 August 2013).
4 Anonymous, Copyright protection for computer programs, Columbia Law Review, 64 (7) (1964) 1274-1300.
6 Raysman R, Protection of proprietary software on the computer industry; Trade secrets as an effective method, Jurimetrics Journal, 18 (4) (Summer 1978) 335-351.
34 Harris J R, A Market-oriented approach to the use of trade secret or copyright protection (or both?) for software, Jurimetrics, 25 (2) (1985) 147-167.
37 “As technology advances, computer-implemented inventions are increasingly being used in all fields of technology, and in many cases the innovative part of a new product or process will lie in a computer program. As a result, the impact of these inventions on our daily lives is growing, and so too is their benefit to individuals and to society as a whole. The amount of R&D resources put into the development and commercialisation of these products is enormous. It is doubtful whether innovators would make such an effort if they did not expect to benefit economically from their work. Patent protection therefore plays a key role in innovation strategies, and not just in the field of computer-implemented inventions. Many important innovations have reached the marketplace with the help of the patent system.” Patents for software? European law and practice, European Patent Office, www.epo.org.
38 “India has a balanced political outlook and an independent judiciary, and in the last decade or so has done well in harmonising its patent law with that of other major jurisdictions. However, software protection is weak and the need to provide stronger protection for software inventions has been the subject of debate both domestically and around the world. Stakeholders agree that stronger protection is
considered as patentable subject matter in the USA”;


70 Computer-implemented inventions (CII): Inventions that work by using a computer, a computer network or other programmable apparatus. To be patentable, they must have technical character and solve a technical problem, be new and involve an inventive technical contribution to the prior art, European Patent Office, http://www.epo.org, (accessed on 12 October 2013).

75 “… practices and case law regarding the patentability of software-related inventions vary significantly in different countries”; http://www.wipo.int/patentscope/en/patents_faq.html (accessed on October 2013).
76 Mark A Lemley and A Douglas Melamed, Missing the forest for the trolls, Columbia Law Review, 113 (8) (December 2013) 2117-2189.
77 Pohlmann T & Opitz M, Typology of the patent troll business, R&D Management, 43 (2) (2013) 103-120.
80 2013 Patent Litigation Study, © 2013 PricewaterhouseCoopers LLP.
83 Servers are called as “Heat Generating Machines”.
84 Parker H, From State Street Bank to CLS Bank and back: Reforming software patents to promote innovation, Vand. J. Ent. & Tech. L, 16 (2) (2014) 425.
86 Opperman C, Software folly Legislating against software patents in Europe will not help smaller companies compete with the multinationals. In fact, it will achieve the opposite. Intellectual Asset Management, 2004, 7-10.