Interface Between US And Indian Copyright Laws In Reverse Engineering Of Copyrighted Computer Programme

A.P.Divya¹, P. Chandran², Dr. Balasaheb Garje³

¹,²Research scholar, Department of Law, Himalayan University, Itanagar, Arunachal Pradesh, India.
³Research Guide, Department of Law, Himalayan University, Itanagar, Arunachal Pradesh, India.

Abstract
Technology is evolving everyday as consumers spend countless amounts of money buying new products and companies compete to produce better products. One catalyst of this technological innovation is reverse engineering by both developers and consumers. Reverse engineering is a method of recreating existing engineering concepts by analyzing the design and components of a final product to ascertain how the product operates. Although this is clearly distinguishable from the traditional concept of forward engineering- which requires creating a product from abstract engineering ideas and concepts- it has been practiced as a useful tool to learn how to build a technology and make improvements. Reverse engineering is well – exemplified in the computer software industry, where programmers constantly examine existing software to better understand the structure and make improvements on its operability. Some people think that it has promoted the innovation and development of the software industry, and it has a great significance when promoting the development of science and technology society, but some people think that there are still some problems with the legality of reverse engineering. Therefore, it is necessary for us to learn from the advanced foreign legislation and make a detailed analysis of the conditions for the implementation of reverse engineering in our country.

Key words: technology, Reverse engineering, Copy right, Software, Computer Program.

Introduction
The legal scheme of copyright is presumptive of the need to extend protection to new and evolving forms of art and creativity. It is needless to mention that a complex technology like computer program has often juggled hard to adapt itself to the copyright law framework. After long and prolific deliberations on the legal and policy implications of extending copyright protection to computer programs, it is now well established that it has grown as a distinct sub-discipline in copyright system across the world. Reverse engineering is an effective tool to drive competition and innovation, to promote a greater good for consumers and the public at large. One way to do this is by providing more flexible interoperability exceptions for reverse engineering to expand choices and reduce cost for consumers.
Copyright protection for computer programs has sound juridical basis in international and national copyright law. The Indian law on copyright - The Copyright Act 1957 (as amended up to date) specifically extends copyright protection to computer programs as a form of literary work. A programmer while developing a program has to work under constraints of programming language and techniques that may fall within the continuum of ideas, expression and algorithms. While ideas are not protected, expressions are.

The scope of what falls within the protectable expression and unprotectable idea is resolved by the application of idea-expression dichotomy. Copyright protects originally in expression. However, it also extends protection to non-literal elements in literary works. This presents various challenges in discerning ideas from original expressions. Protection for computer programs can embody the interface within it. Such interfaces (both user and functional) are copyrightable written expressions and utilitarian functionality (ideas). Such functionality must be easily accessible to all since they form ideas in a program. The Indian courts have invoked the idea-expression dichotomy from time to time as having doctrinal basis in the Indian copyright system.

While the debate on the rationale of extending copyright protection to computer programs seems to have fairly settled, there still exists a murky legal area concerning the nature, scope and extent of fair dealing of computer programs. Much of the concern is due to the nature of innovation in software technology and the compatibility requirements of computer program, technically referred to as achieving interoperability. This essentially harps on an important point that computer program are utilitarian works and hence face special problems. As functionality in a computer programme is not protected by copyright law. Computer program that de facto cover functionality must be allowed for analyzing their underlying idea. The fact that the object code (which cannot be understood by man) is protected through copyright, determining ideas/function or achieving interoperability through the process of reverse engineering without incurring the guilt of infringement is impossible.

1. Art.10.1 TRIPs Agreement
2. Sec 2(o) of the Copyright Act 1957.
4. Art. 13 TRIPs Agreement

Much of the concern regarding decompiling emanates due to intermediate copying of a program. The reproduction, adaptation and translation rights accruing to the author are generally in question. Reverse engineering methods involve copying of the copyrighted computer program and the question is if such intermediate copying as a procedure is legally sustainable in consequential to the final outcome. Thus intermediate copying (unless excused for specific purposes) is violative of the reproduction right of the copyright holder. Intermediate copying during reverse engineering, however is not straightforward copying since the output of the reverse engineering code may be non-infringing and hence should be allowed under the theory of fair use. Further, the economics of innovation in the software industry demands that
reverse engineering through de-compilation should not be prohibited\(^8\). Some have even argued that even if reverse engineering does not lead to additional innovation by building competing products it does provide consumer with an equal choice at a competitive price and hence enhances consumer welfare. Computer program technology presents a strong possibility of swathing unprotected elements of a program that are beyond the scope of copyright and hence must be available in the public domain. The necessitates access through principles of fair use. Consequently, the copyright act 1957, itself provides for certain exceptions in the nature of reverse engineering and other limited exception. However, private ordering through contract law that imposes restrictions on fair dealing/use can undermine the essential balance that the copyright law exceptions provide. The object of this paper is to examine to what extent the reverse engineering of computer program is permitted under the Copyright law to make compatible, interoperable and innovative program. Further to examine the legal validity of the shrink wrap license to the extent prohibiting reverse engineering. This paper analyses the nature and scope of reverse engineering of computer program to what extent reverse engineering is permitted under the copyright Act as a fair use. The legal validity of the contractual term that imposes restriction on reverse engineering and the provision of the Competition Act and anti reverse engineering clause cause or likely to cause adverse effect on competition and development in software industry.


**Nature of Reverse Engineering of Computer Programme**

Reverse engineering is a process for understanding the structure and functionality of software, given the source code. Thus reverse engineering is often employed for understanding new innovation by a competitor and achieving compatibility between two products or programs. Goals of reverse engineering include developing competing software or hardware peripherals and compatible new products, uncovering the weakness of products, and satisfying idle curiosity\(^8\).

In the typical software development process, computer programs are initially written in an alphanumeric language that consists of words and arithmetical expressions meaningful to humans (source code). The source code is then translated or compiled by a utility program into a computer-readable code (object code). Reverse engineering involves obtaining either the original source code or detailed written specification from the original programmer. Otherwise it must be undertaken independently by decompiling the object code back into the source code. As a practical matter, it may be impossible to reverse engineer a computer program without decompiling the object code back into an equivalent source code version.\(^9\) With respect to fair
use argument in copyright law, it is significant to understand that the process of decompilation generally cannot be achieved without at some point making copies of the original program.  

**Reverse engineering is not inherently unlawful:** The process has been occurring for many decades within the industrial and scientific communities. Its legality under copyright law depends on the particular purpose of reverse engineering. The US Supreme Court has defined reverse engineering as a “fair and honest means… of starting with the known product and working backward to divine the process which aided in its development or manufacture”.


In this respect motivation and contexts of software reverse engineering are divided into two major categories: developing non competing vs. competing programs. Within this distinction, there are many sub distinction including making copies for using part or all protected features of the original program, error correction, improvement, functional equivalents, and inter of interoperability. Making a copy is a prima facie case of infringement because the initial result of the process of reverse engineering is substantially similar to protected work. When it comes to the purposes of functional equivalents and interoperable products, however the fair use defense for intermediate coping occurring de compilation may be considered.

Historically, most computer programs were designed to perform independently from other programs making interoperability less important than it is today. However, technology advances the value of a computer program to its users while also making it increasingly dependent on compatibility or interoperability with a particular hardware or software system. Understanding a protected computer system or operating programs by using the technology of reverse engineering is obviously necessary for new application development and competition. In reality, however, a copyright owner of an operating system or software application expends great efforts to prevent reverse engineering of its program by its competitors or third parties by way of using copyright and creating technical tools such as a password or “lock-and-key” programs. Such situation give rise to the most contentious legal question in recent decades: how the “fair use doctrine” resolves the underlying intermediate copying during the de-compilation of a copyrighted program for the purpose of compatibility with the program. Furthermore, under what circumstances should de-compilation be allowed as a legitimate use within copyright protection?


14. Vault Corp. v. Quaid Software Ltd., 847 F. 2d 255, 256 (5th Cir. 1988)

There are roughly four methods of observation and analysis under the umbrella of software reverse engineering; (1) read about the program in the manuals, (2) observe the program in operation by loading and running it on a computer and viewing what the program does on the screen displays, (3) perform a static examination of the individual computer instructions contained within the program, or (4) perform a dynamic examination of instructions as the program is being run on a computer. 15

The fourth method of reverse engineering is pertinent to de-compilation of the object code back in to source code. Engineers use a program known as a disassemble (or de-compiler) to translate the zeros and one of binary machine readable object code into the words and mathematical symbol of source code.

Scope of Copyright Protection for Reverse Engineering of Computer Program

Idea v. Expression Dichotomy of Computer Programme

Copyright law does not protect ideas, but protects only the expression of ideas. This feature of the copyright law acts as a severe limitation with respect to the protection of computer program since in the case of a computer program which performs a particular function achieves a specific task, others will be permitted to create their own program to perform the same function as long as they do not copy the form (literal elements) of the earlier one. However most of the countries consider copyright as the most appropriate form among the two, because they feel that copyright secured the best compromise between protection and competition. Computer programs are thus globally protected under the copyright law as literary works since this is the international mandatory norm set in the TRIPS Agreement. Initially, there was confusion as to the copyright ability of computer programs since it was argued that at least when the program reaches the electronic form, it becomes a mean of operating the machine and is no longer a subject matter of copyright protection. In Apple Computer Inc v. Franklin Computer Corp., it was held that a computer program, whether in object or source code, is a literary work and is protected from unauthorized copying, whether from its object or source code version. In Whelan associates Inc. v. Jaslow Dental laboratory Inc., the US Court of Appeals for the third Circuit attempted to tackle the problem by suggesting that the purpose or function of a computer program was the idea of the program as a whole, and everything that was not necessary to that purpose or function would be part of the expression of that idea.


In Lotus Development Corp. v. Paper Back Software International\textsuperscript{17} the issue of copyright ability of non literal elements in computer program was considered in detail. The court expressed the view that when computer programs include elements both literal and non literal that can be identified separately from, and are capable of existing independently of, the utilitarian aspects of the article, they are potentially copyrightable.

Another major problem which needed to be resolved with respect to copyright ability of computer programs was regarding the separation of ideas from expression. This was somewhat effectively done in Computer Associates International Inc. v. Altai Inc \textsuperscript{18}. The court in that case used abstraction, filtration and comparison test for separating ideas from expressions.

In the abstraction – filtration – comparison analysis, the court first broke down the allegedly infringed program into conceptually separate modules. By examining each of these parts for such things as incorporated ideas, expressions that are necessarily incidental to those ideas, and elements that are taken from the public domain, the court would enable another court to sift out all non protectable elements. The court applied the doctrine of merger and the scenes a faire doctrine to computer software at this filtration stage. It held that the merger doctrine precluded copyright protection for the particular set of modules, which was necessary to promote program efficiency or speed, in such instances where its idea could not be separated from the creative and technical expression. As the final step, the court isolated the golden nugget protectable code and compared this with the software application at issue. The result of this comparison determined whether the protectable elements of the programs were substantially similar so as to warrant a finding of infringement. This better approach has been approved by other courts. Though the Indian Copyright Act, 1957 started protection of computer programs as early from 1984, unfortunately there are no reported cases in Indian on the above issues.

\textsuperscript{17} 740F Supp37: 15 USPQ 2d 1577.

\textsuperscript{18} 740 F Supp 38: 15 USPQ 2d 1577

**RESEARCH STATEMENT**

Reverse engineering right is necessary to strike a proper balance between providing adequate copyright protection and promoting public access to a copyrighted work. A major fair-access issue has arisen in situations where a copyright holder attempts to restrict a reverse engineering right by copyright infringement litigation, contractual shrink-wrap licenses, or protective system even though the reverse engineering is permissible under the “fair use” doctrine. Further, if a software vendor that has market power in the primary market intends to eliminate existing or potential competition by unilaterally imposing anti-reverse engineering terms, and in doing so brings about anti-competitive effects in the relevant markets. Importantly software industry represents strong network effects. One convincing argument is that network effects support permitting a reverse engineering right in order to achieve
compatibility with a dominant entity since dynamic competition benefits from network effects, which will increase consumer welfare.

**RESEARCH METHODOLOGY**
Since the study is doctrinal base, the author relied on primary and secondary sources like international instruments, law, books, journal, and web source. Analytical method is used for the critical analysis of copyright, ant circumvention, and competition law to reach the goal. Further, the author made an analysis of the international instruments of US, EU and India on computer program by employing comparative method.

**OBJECTIVES OF THE STUDY :**
1. To study the scope of copyright protection of computer program.
2. To critically analyze the TRIPS Agreement and legal provisions in EU, US and India on computer program and competition.
3. To understand the relevance of reverse engineering in the development of compatible or interoperable program, competitive program and enhancement of programs.
4. To examine the provisions concerning fair use/dealing of computer program in EU, US and in India and critically evaluate its nature and scope.
5. To study the impact of Anti-Circumvention law and the scope of Competition Act on the reverse engineering of computer program.
6. To examine the legal validity of software license [EULA] prohibiting reverse engineering.
7. To critically appraise the judicial decision on idea-expression dichotomy in computer program.
8. To study the doctrine of ‘copyright misuse’ and to understand the ‘essential facility’ doctrine and its relevance to prevent the abuse of monopoly right by copyright owner.

**EXAMINATION OF QUESTION:**
1. Copyright protection of computer program conferring on the copyright owner a de-facto monopoly on functional aspects result in patent like protection.
2. Reverse engineering serving valuable social purposes may pave the way for protection from liability.
3. The impact of the Anti-Circumvention Law on non-infringing uses of copyrighted work leads to friction in the objective of copyright law.
4. Contract terms aiming to override the decompilation privilege would prevent the innovation in software industry.
5. The anti-competitive clause in software license agreement results in copyright misuse.
6. Copyright protections of computer programs were designed to promote competition to restraint the dominant company monopolizing in the relevant market.

**The Fair Use Defense for Reverse Engineering of Computer Programme**
The U.S. Copyright Act may excuse a reverse engineering from copyright infringement liability if she or he is engaged in a fair use of copyright protected works. The fair use doctrine allows others to use a copyrighted work without the owner’s consent in a reasonable manner for certain purposes such as criticism, comment, news reporting, teaching, Scholarship or research. Copyright law confers upon an author certain exclusive rights. However, some
limitation exists on the author’s exclusive right to the copyrighted works, since the public benefits from wide dissemination of creative works. Among these limitations, the most important is known as “fair use.” An intermediate copy which is made during the process of reverse engineering in order to understand the ideas and functional elements embodied in the copyrighted computer program may be considered to have a “research” purpose as a fair use exception.

In *Sega Enterprises Ltd v. Accolade, Inc* 21 the US Court of Appeal for the Ninth Circuit faced the difficult question of whether the fair use doctrine should be applied to the case of disassembly of computer software in order to achieve compatibility with a copyrighted word. Sega Enterprises Ltd., the plaintiff, and its subsidiary, Sega or America (collectively referred to as Sega), developed and marked video game systems, including the “Genesis” console and its corresponding video game cartridges. Accolade, Inc., the defendant, is an independent developer of computer entertainment software, and desired to make video game cartridges that would be compatible with the Genesis game system.

Sega licensed its copyrighted computer code and its SEGA trademark to a number of independent developers of computer game software. Those licensees manufactured and sold Genesis-compatible games in competition with Sega’s game cartridges. Accolade sought a licensing agreement with Sega, but since the agreement would have required that Sega be the exclusive manufacturer of all games produced by Accolade, it chose not to be a Sega license. Instead, Accolade “reversed engineered” the microcode contained in Sega video game cartridges by using a process of decompiling the binary object code in order to discover the method of interoperating with the Genesis console.

The court concluded its copyright infringement analysis by stating that “where disassembly is the only way to gain access to the ideas and functional elements embodied in a copyrighted computer program and where there is a legitimate reason for seeking such access, disassembly is a fair use of the copyrighted work as a matter of law.”

Indian copyright Act 1957 (Amended by 1999) aims to provide specific reverse engineering exceptions for interoperability and program analysis and also other fair use exception. Sec 52(1) (aa),(ab), (ac) and (ad) reads as follows:

(aa) the making of copies or adaptation of a computer programme by the lawful possessor of a copy of such computer programme, from such copy-

(i) In order to utilize the computer programme for the purposes for which it was supplied; or
(ii) To make backup copies purely as a temporary protection against loss, destruction or damage in order only to utilize the computer programme for the purpose for which it was supplied;
(ab) Doing of any act necessary to obtain information essential for operating interoperability of an independently created computer program with other programs by a lawful possessor of a computer program provided that such information is not otherwise readily available;
(ac) The observation, study or test of functioning of the computer program in order to determine the ideas and principles which underline any elements of the program while performing such acts necessary for the functions for which computer program was supplied;
(ad) the making of copies or adaptation of the computer program from a personally legally obtained copy for non commercial personal use;”

However, the Indian Act does not use the word reverse engineering or de-compilation but it states that the legal possessor of the copy of the computer programme can do any act for the purpose of obtaining information necessary for the purpose of interoperability. It is clear that Indian Copyright Act permits reverse engineering of computer program and promotes competition.

**Contractual Restrictions on Reverse Engineering**
Traditionally, U.S. contract law has governed the transfer and exploitation of materials protected by copyright law, as with any other statutorily created property rights. Hence they are separate branches of law- the contract law used for private ordering and the copyright law seen as more of a statutorily created bundle of rights. However, recent years have seen drastic changes in the way copyright and contract law share their relationship due to the emergence of new models of distribution of copyrighted works and as corollary to changing market practices necessitated by possibilities unfolded by new generation technologies.22 Private ordering and technological self-help of enforcing copyright through contracts has gained prominence in view of the fact new form of digital rights are presumed to help the copyright owners to exercise more control over copyright works produced by them.23 It is presumed that private contracting of property right like copyright to place restrictions on statutorily created exceptions and limitations will grant an incentive that copyright law is any way bound to further.

It is well understood that the use of mass market End User License Agreement can fundamentally alter the balance of interests recognized under copyright law and policy.24 A mass-market transaction is a consumer transaction and any other transaction for information or informational rights directed to the general public as a whole under substantially the same terms for the same information with an end-user license. The mass market contracts are not contract between a few individuals, but rather for the general public. Private contractual arrangement can be facilitative of copyright holder’s willingness to restrict reverse engineering or restrain fair use. “Shrink-wrap and “Click-on”25” are the two common modes to place contractual restriction on reverse engineering, where they demand adherence of such contracts as a condition of licensing software.


24. Supra note 27

25. A click-wrap license is the electronic equivalent of shrink wrap licenses. This type of license appears on a user’s computer screen when the user first loads a computer program, and requires the user to click his acceptance of the terms of the license before he uses the software.

**Extra Element Test**

The validity of such licenses as being contrary to fair use was challenged in many cases in the US, but the law is still unsettled, the primary argument raised in the US context is that state contract law that allows contractual restraints on reverse engineering are not preempted by the federal copyright law. It is argued that fair use is a district claim and can be contracted out by copyright holder. In the context of decompilation of computer programs, contractual restraints is an issue if by enforcing such contracts affects or places restriction on fair use defense. To address this inherent conflict between contractual restraints and copyright fair use or other exceptions, the US courts have generally used something called the “extra element test or additional element test.”

This test asks whether a state law right requires proof of an extra element that is qualitatively distinctive from the federal right. Preemption may be denied if one or more qualitatively different elements, which constitute the state-created cause of action, are found in addition to those required for copyright infringement. In Bowers v. Baystate Tech. Inc., the issue was whether the copyright law preempted the prohibition of reverse engineering through shrink wrap licenses.

The court held that Copyright Act did not preempt the plaintiff’s contract claim. It found that reverse engineering was an “extra claim” that the parties were free to negotiate through a contract. It appears that the court was too much emphatic about freedom of parties to contract. It went on to muster support for its reasoning from some earlier case developments, specifically on ProCD, Inc. v. Zeidenberg, jurisprudence. In the ProCD case, the court had stated that the federal copyright law did not preempt the state trade secret law since the additional elements of proof required in trade secret law were enough to make a claim qualitatively different from the copyright claim. Thus the Bowers court took shelter to uphold contractual claims that prohibited reverse engineering. The court also placed reliance on a different circuit court judgment, a similar case of “Contractual reverse engineering restrictions” in which it was held that a simple two party contract was not “equivalent to any of the exclusive rights within the general scope of copyright” and was therefore not preempted by federal copyright law.

26. Alcatel U.S.A. Inc. v. DGI Techs. Inc., 166 F. 3d 1447, 1454 (7th Cir. 1996)


28. Ibid
However, some criticism of this test indicates that this decision is arbitrarily made by courts rather than on the basis of any real guidance: “there is always some difference between the state law and the Copyright Act”\textsuperscript{29} If a court wants to avoid preemption, it can always find some difference that becomes the “extra element” needed to avoid preemption. In contrast when there is preemption, no “extra element” is put on the label.\textsuperscript{30}

**The Supremacy Clause**

The supremacy clause of the Constitution provides a more feasible ground for pre-emption questions arising out of the application of the Copyright Act.\textsuperscript{31} In determining whether a state statute is pre-empted by federal law and therefore invalid under the Supremacy Clause of the constitution, the test also relies on ascertaining the intent of Congress. Federal law may supersede state law in several different ways: first, Congress has mandated by express decree; second, “the scheme of federal regulation is sufficiently comprehensive to make reasonable the inference that congress ‘left no room’ for supplementary stat regulation;” and third, a state law conflicts with federal law when “the state law stands as an obstacle to the accomplishment and execution of the full purposes and objectives of Congress.”\textsuperscript{32}

Congress grants certain exclusive rights or monopoly privileges to creators, but at the very same time, courts and legislatures have seen the need to limit the scope of copyright monopoly to strike a copyright balance. The copyright principles and doctrines such as the fair use, first sale doctrine, the idea expression doctrine and the copyright misuse doctrine reflect this need for limitation on copyrights. These limitations are an important step toward achieving a uniquely constitutional goal: promoting the progress of science and the useful arts. As one of the “built-in accommodations” contained in copyright law, the fair use doctrine furthers this constitutional goal, which “allows authors and others to bring to market expressive works of potentially great public benefit that are made possible only through the fair use of another’s expressive work.”\textsuperscript{33}

\textsuperscript{29}86 F. 3d 1447 (7\textsuperscript{th} Cir. 1996).


\textsuperscript{31}Schuyler Moore, Straightening Out Copyright Preemption, 9 UCLA Ent. L. Rev. 201, 204 (2002).

\textsuperscript{32}The Supremacy Clause provides that the “Constitution and the Laws of the United States which shall be made in pursuance thereof… shall be the supreme Law of the land; and Judges in every state shall be bound thereby, and thing in the Constitution or Laws of any state to the contrary notwithstanding”.

While the right of software users to reverse engineer is not established by statute, no federal court of appeals has disagreed with the notion that reverse engineering constitutes fair use in certain circumstances. Fair use activities often could result in the creation of new and useful software products. Clearly, in light of the constitutional goal of promoting the progress of science and the useful arts through the balance established in the Copyright Act, a prohibition of such activity would stand as an obstacle to the full purpose of Congress. Thus under the supremacy clause analysis, such restriction on reverse engineering should be preempted.

In Vault Corp. v. Quaid Software Ltd, the US Court of Appeal for the Fifth Circuit applied a supremacy clause analysis to set aside a contractual restriction on reverse engineering, noting that the provision “conflict with the rights of computer program owners under s117 and clearly ‘touches upon an area’ of Federal Copyright law.”

**Doctrine of unconscionability**

Contracts terms prohibiting reverse engineering may be unenforceable because they are unconscionable. The basic test is whether contracts or clauses involved are as one-sided and unfair as to be unconscionable under the circumstances existing at the time the contract was made. In general, unconscionability cases involving bargaining misconduct can be decided under the standard contract law defenses, such as fraud, duress, undue influence, mistake, inability, take it or leave it negotiating postures, or illegality. Contract of adhesion are offered on a non negotiable take-it-or-leave-it basis by a party having superior bargaining position.

A shrink-wrap license restricting reverse engineering may be void if imposed in a take-itleave-it option, but will be enforced if embodied in an agreement reflecting deliberately assent in a commercial setting. In Vault Corp. v. Quaid software, ltd. the court stated that a shrink-wrap license is unenforceable as a “contract of adhesion.”

---

33.Ibid
35. 847 F. 2d 255 (5th Cir. 1988).
36.Intel Corp. v. Intergraph Corp., 195 F.3d 1346, 1365 (Fed. Cir. 1999)
37.Supra n. 40

A licensor of the adhesion contract in the position to refuse to bargain for some reason gave a licensee a take-it-or-leave-it option. In the Indian context, the Copyright Act, 1957 does not explicitly provide for contracting out “fair use” exceptions clearly provided under the Act. However, it does not even explicitly bar such acts. Since there is no case law on this point, it is unclear if contractual provision will override the statutory provision expressly permitting fair use. Even the statement of object and reasons does not provide any practical guidance on this crucial point. Considering that public access to copyrighted knowledge has been cornerstone of Indian copyright law, any contract that prohibits fair use- which is central to copyright law- is prima-facie void. There is no overriding public policy reason as to why such contract must be validated.
Prohibiting reverse Engineering and competition law

Any agreement between enterprises, person or group of persons with respect to supply of goods or services which causes or likely to cause an adverse effect on competition within India is void as stated in Sec 3 of the Indian Competition Act, 2002. This section will not apply to any restriction that the owner of any intellectual property rights may impose, in the exercise of his rights, to restrain infringement of any of his rights, or to the right of any person to export goods from India to the extent to which the agreement relates exclusively to the production, supply, distribution or control of goods or provision of service for such export. The scope of the exception is narrow if the owner of the Intellectual Property rights imposes a condition on their licenses that go beyond the limits of the rights conferred on them by law and which conditions may reduce competition or actually eliminate competition in the relevant market is prohibited under the Act.

Computer program protected as a literary work under the copyright is different from the other literary work. It is a useful article; the work itself carries the underlying idea within it which cannot be discerned without reverse engineering. The law itself permits the doing of reverse engineering to know the functional element in a computer program for the purpose of promoting the development of interoperable and compatible product. But the owners of computer program restrict the same by imposing a condition in the license term prohibiting reverse engineering. This term has to be strike down as anti competitive in nature under the per se rule causing adverse effect on competitive market.

38. Sec 3(5) of the Indian Competition Act 2002.
39. Sec 52 of the Copyright Act 1957.

The computer program part of “network externalities” each program has to be developed in such a way that it runs with other program. Application program has to be compatible with operating system software and with other application program the value of one will depend on its compatibility for that underlying interface should be known that is possible when the source code is available.

Most of the programs available in the market is in CD form which carries the object code (not a human readable language) and the same is protected under the copyright law if we make a copy it amounts to infringement. Law permits intermediate copying to promote the development of original computer program the same cannot be taken away by the owner of Intellectual property by prohibiting reverse engineering. We have to wait and see how the Judiciary is going to tackle with this issue in the future.

Limitations of the Study

The study is confined to copyright protections of computer program and that to restrict to computer software distributed only in its unintelligible object code, so called “closed software”. The author set aside open source software for the purpose of PAPER. In the techniques of de-compilation (or disassembly) or conduct an analytical “clean room” operation process, a program’s object code is first copied and then transformed into a human readable computer code, the source code. In addition to intermediate copying of the original computer program (de-compilation), the process of reverse engineering also involves building the source code program based on the decompiled information of
the original object code (implementation). Decompiling object code produces an approximation of the original source code, known as “intermediate copy

CONCLUSION AND SUGGESTIONS
In the 1950s, the computer industry has been targeting the general public, rather than the individual author of a computer program a communitarian view of software development was the norm, and programmers openly shared their ideas with one another. As the technological zeitgeist took its pace in the world economy, software developers recognized the vast profitability of software, and they began to exercise tight controls over their computer codes by different means. However, most of the countries consider copyright as the most appropriate form of protection of computer program, because copyright secured the best compromise between protection and competition. Computer program are thus globally protected under the copyright law as literary work and this is the internationally mandatory norm set in the TRIPS Agreement.

Computer program which performs a particular function and achieves a specific task, others will be permitted to create their own program to perform the same function or compatible program, as long as they do not copy the expression of the earlier one. Computer program are typically distributed for public use in object code form, embedded in a silicon chip or on a floppy disk. For that reason, human often cannot gain access to the unprotected idea and functional concept contained in object code without disassembling or decompiling that code i.e., making copies. Unless an explicit exception recognizes copying of code would violate the programmer’s exclusive rights. This will be true, even if the person who decompiled the code only used it as an interim step in gaining access to the programs functionality or developing a new, independent computer program. As reverse engineering including decompilation, has become a common practice particularly in software industry. To legalize the copying of code to access functional elements recognized the reverse engineering as a fair use. The public policy behind the fair use doctrine is important when analyzing reverse engineering issues in conjunction with another characteristic in the high tech industry namely innovation accomplished through a succession of incremental improvements. The resulting mutual dependence of manufacturers yields a ‘network effect’. Network effect greatly affects the development of software products if a large number of compatible products- with an industry standard possessed by a dominant market entity are available in the market. Software products produced by competitors often must have compatibility with an industry standard because they benefit from the network effects of the standards widespread adoption. It permits competition among providers, which also benefits the market. Therefore effective competition requires both computer software and hardware developed by competitors to be compatible with an existing standard system. The reverse engineering right in network industry should be allowed in order to accomplish compatibility. Innovations in the network industry are less of a public good if a large number of distinct participants cannot employ the ideas simultaneously. Thus defining the scope of access right to the protected work is important. As a common law tradition nation, federal jurisdiction in the United States tried to settle the de-compilation battle via fair use tactic. Despite the fact that the balancing test of the fair use factors under section 107 of the
Copyright Act leaves some discretion to U.S. judges, the parties involved in the de-compilation dilemma are caught in a “no win” situation in either way.

On the one end of the spectrum, an extension of the fair use doctrine into the area of de-compilation would leave software providers powerless to stop the use of their software by others, especially competitors, to plunder profits that would otherwise belong to the software providers. At the other end of the spectrum, without the extension of the fair use doctrine into the area of de-compilation, software providers would be able to receive copyright like protection of functional elements embodied in their software due to the fact that the utilitarian aspects of the software are not directly observable. The European Communities have ventured to establish an identity in the global software world, especially to be able to form a counter-part to the U.S. dominance in the software market. These efforts culminated in 1991 in the enactment of the EU Software Directive with the highly controversial de-compilation exception of Article 6. By adopting a reasonable devised exception for decompiling 460 computer program Europe has come up with a worldwide unique model of how to walk the thin line between public and private interests in the reverse engineering software industry. The de-compilation exception may be cited as manifest for Europe’s position in the de-compilation battle, as one towards “open systems”. Article 6 centers around interoperability as the indispensable criterion to lawful software de-compilation under European Copyright Law. Interoperability serves as legitimate reason for decompiling computer software permissible under copyright law. On the other hand, software interoperability functions to ensure competition in a market dominated by a monopolist, like an owner of copyright in its computer program. Thus, interoperability is the key criterion in both copyright law and competition law involving software de-compilation. Although the objectives pursued under copyright and competition law are different, the remedy of compulsory licensing of essential facilities (such as interface specifications as basics for accomplishing interoperability) may be efficient to the same extent in either law. A mandatory licensing order in pure copyright cases would effect to bridge the inherent conflict of software copyright and de-compilation in a way as would be unique in the world copyright law regimes. However, thus far it has been an issue exclusively for European competition law to determine to which extent competition law can be used to force compulsory licensing of intellectual property rights. Beyond the traditional role of the compulsory licensing remedy in competition law, mandatory licensing could also serve as remedy to reconcile the contradicting private and public interests in the software de-compilation context.

In Microsoft, a case settled in European competition law, the CFI aimed at reconciling Microsoft’s Interest in its intellectual property and the general public’s interest in intellectual growth, which is one of the objectives of the EU Software Directive, by way of compulsory licensing beyond the traditional scope of copyright law. The same imbalance has occurred under U.S. copyright law, though solved in a different way. The Ninth Circuit, in Sega and Sony, stripped of the copyright incentives and rewards that first had coaxed software providers into producing and disseminating their works, while thereby preserving competition. Harsh 461 consequences, like in Sega and Sony, might have been avoiding by imposing on them the obligation to enter into reasonable licensing agreements with Accolade and Connectix.
Generally, the current frustration among U.S. Software providers could so be mitigated in two respects: First by way of compulsory licensing Sega and Accolade would be financially reimbursed for their creative efforts, skill and resources spent in software programming. Second, the practice of compulsory licensing clearly expresses the recognition of the software providers’ copyright and their protection there under. Both these reason may contribute, on the one hand, to strengthening the confident of software providers in the copyright protection of their computer software. Nonetheless, compulsory licensing as an alternative to de-compilation in the United States, for the time being, has not been a substantial issue. In response to the competitive implications of software de-compilation U.S courts cautiously apply the copyright misuse defense. In the face of Magill, IMS Health and Microsoft, there is a perception that the Commission considers that the protection afforded by national intellectual property rights is prone to go beyond the legitimate reward for the software author’s creative and financial efforts. It is in cases involving intellectual property rights where there is the greatest danger of misuse of an essential facility. While U.S. software providers may invoke copyright misuse defense, European case law-devoid of a copyright misuse doctrine or alike –refers to the essential facility doctrine for software providers to apply in cases where national copyright laws provide for near–monopoly protection up to an anti-competitive extent. Under European law, it is competition law that may oblige a dominant owner of the essential facility, including computer code, to operate with its competitors, on competition grounds. In so doing, courts have relied on basic principles of antitrust economics of the impact of refusal to deal. A similar balancing test, which is yet neither undisputed nor uniformly applied by U.S judges, has been established under U.S. case law when referring to the doctrine of copyright misuse.

Today, the misuse defense has not been codified in the Copyright Act. Neither has the U.S. Supreme Court yet clarified the existence and scope of copyright misuse. This reluctance indicates another concession to 462 enhanced protection of software copyright owner vis-à-vis reverse engineers, and thereby continues to pursue a “closed source code” model. At the other end of the Atlantic, the European compulsory license approach as adopted in Microsoft, from a copyright standpoint, proves efficient likewise in that compulsory licensing of Interface specifications for the purpose of interoperability may contribute, on the other hand, to strengthening the confident of software developers in the copyright protection of their computer software. So the unwanted side-effects triggered off by “all-or-nothing” decisions, such as in Sega and Sony could then be avoided in future. In affirming the Commission’s decision in the Microsoft case the CFI appears to reinforce Europe’s Position in the software industry, particularly vis-a-vis dominant U.S Software providers. By responding to the refusal of access to interface information contained in copyrighted computer software with a compulsory licensing order the European Communities would give further direction to an “open software ”model. After all, the author advocates that compulsory licensing would be the most “open” solution in the de-compilation context because it would allow persons who engage in software de-compilation unrestricted access to another program’s interface specifications upon the payment of a compulsory fee as reimbursement for the software provider’s efforts in the development of that interface. Furthermore, compulsory licensing in copyright cases under the auspices of the judiciary would likely deter reverse engineers from deliberately seeking access to copyrighted computer code for purposes of subsequent software piracy. However,
this research questions whether the United States will follow the Microsoft approach given the U.S copyright law as evolving at different settings in to a “closed software” attitude. With the transformative approach adopted under Sony the permissibility of de-compilation (and reverse engineering) has been limited to the creation of only non-competing programs. Reimerdes, a case decided under the DMCA, leads to an ambit of the reverse engineering exemption narrower than the fair use doctrine because the court in finding copyright infringement focused on the early stage of intermediate copying in the overall process of reverse engineering. Finally, there may be observed a trend under contract law according to which anti-463 de-compilation (and anti-reverse engineering) clauses are deemed enforceable with the effect that minimum de-compilation rights may be contracted away in private stipulations by the U.S. Software author.

The Indian Copyright Act, 1957 through section 52 (1) (aa) (ab) (ac) and (ad), does take care of such a situation to larger extent. The scope of permissibility of decompilation has been limited to the creation of an original program and information essential for interoperability is not readily available. Further the Act is silent with respect of interoperability with the hardware component of a computer. The fair use doctrine acts as a safety valve in permitting reverse engineering of computer program to access the interface. But the situations become worsen when the owner of the copyrighted work started using the TPM to prevent access to the copyrighted work. The use of these new technologies also created certain unintended consequences like it restricts the use of the work which is otherwise possible in the digital context creates problem of interoperability. Anyone who circumvents the technological protection measures he will be liable for the circumvention. In India based on the Anti Circumvention Law passed Copyright Amendment Act 2012 added a new section 65 A. But sec 65 A needs certain clarification as to whether access control TMs or certain specific activity controlling TMs or both come within the definition of “effective technological measures”. The Amendment has not defined the term TPM and what do you mean by circumvention of technological measure. This must also be legislatively clarified and to this effect the provision has to be amended. The exception to section 65 A (1) allows circumvention of technological measures in all cases where such circumvention is not prohibited under the Act. Fair dealing of computer program for various purposes under section 52 (1) (aa) (ab) (ac) and (ad) are acts explicitly provided under the Act and hence place no prohibition within the context of section 65 A (1) exceptions. The only obligation concerning the use of this exception is mentioned in the proviso regarding maintaining a record of persons facilitating such circumvention. It should logically follow that existing interpretations developed pre DRM context shall prevail to the extent of allowing current uses in the post – DRM context as well. 464 It is doubtful of the wording in section 65 A (1) covers the protection against circumvention of both access and copy controls. In that case, the Section does not apply to making of circumventing access controls for the purpose of gaining access to a computer program, but intermediate copying shall be subject to this section. Alternatively, even if access control is interpreted to be within the scope of this section, the exception mentioned therein would still excuse it for all fair use purposes. However, much is left to be desired from the case law jurisprudence will develop in the Indian context.
Reverse engineering is permitted under the fair dealing for the purpose of interoperability and compatibility of independent computer program developed by the lawful possessor. This right is further preserved in the DRM for the purpose of interoperability and beyond, provided the end product is not infringement of the original program. However, a pertinent question is if contract can bar the exceptions created for technological circumvention under the Indian Act. The wording is used is “Nothing in sub-section (1) shall prevent any person from: (a) doing anything referred to therein for a purpose not expressly prohibited by this Act”. As far one possible interpretation goes any contractual restriction of the use of exceptions to technological circumvention is barred because these are acts not prohibited by the Act in fact explicitly allowed. Software is distributed in the market by way of license. Can the licensor i.e., the copyright holder can restrict the reverse engineering by way of contractual agreement. Shrink wrap and click on are the two common modes to place contractual restrictions on reverse engineering, where they demand adherence of such contracts as a condition of licensing software. In the Indian Copyright Act, 1957 does not explicitly provide for contracting out “fair use” exceptions clearly provided under the Act. However, it does not even explicitly bar such acts. Since there is no case law on this point, it is unclear if contractual provision will override the statutory provision expressly permitting fair use. Even the statement of object and reasons does not provide any practical guidance on this crucial point. The copyright holder by way of software license imposes a condition restraining the reverse engineering of computer program is likely to prevent the competitor from developing interoperable and compatible program. Also such a license, prohibit the competitors from accessing the necessary interface to create new and improved program. This would lead to abuse of dominant position by the software giant. Such a restriction exceeds the exception granted under section 3(5) of the competition act resulting in the abuse of exclusive copyright. The commission is posed with a question whether such conduct should be deemed to violate section 3 of the competition Act. More specific by interpreting section 3 (4) (d) which deals with “refusal to deal”1 states that any enterprise enter in to any agreement restrict the person to whom the license was granted from doing any act to understand the necessary interface is considered to be anti competitive and in contravention of section 3 of the Act. U.S. and EU came out with a new doctrine of copyright misuse and essential facility as a solution to provide access to underlying ideas and functional element. The copyright law does not provide any solution for ensuring access to copyrighted computer program, but competition law in the form of essential facility doctrine envisaged under the clause ‘refusal to deal’ ensures a viable alternative to de-compilation for competing software developers whom access to another program source code is denied and therewith access to the relevant market at whole. In other words, perhaps the essential facility doctrine can be viewed as a method for imposing a duty to share one’s copyrighted interface specification for the purpose of achieving interoperability, in cases in which copyright law does not, but should, impose such an obligation, that is a compulsory licensing order. The policy maker should adopt doctrine of essential facility to provide access to interface needed for interoperability. Sec 3(5), the section should go on with a rider to the effect that however, in the exercise of those rights, they would be subject to the relevant provision of the Competition Act. As the section stands, anticompetitive practices in licensing technology would fall under section 3(1), as subsection (3) and (4) deal with specific groups. 1 Sec. 3 (4)(d) Explanation for this subsection – “(d)
“refusal to deal” includes any agreement which restricts, or is likely to restrict, by any method the persons or classes of persons to whom goods are sold or from whom goods are bought”. India may draw lessons from the application of the concept of refusal to deal and the essential facilities doctrine in developed countries in US and EU. If there is refusal to provide access to interface needed for promoting competition. Compulsory license can be used as a weapon, both in IPRs and Competition Law, to remedy anti-competitive practices. Suggestions The author based on the paper forwards the following suggestions for the consideration of the policy makers, to ensure reverse engineering of copyrighted computer program for the purpose of interoperability and to promote innovation and competition in the software industry. 1. Novel and broader interpretation to sec 52 (ab) from a public policy perspective: This can be done, if the policy maker expressly states in the clause that ‘reverse engineering and extreme form of reverse engineering i.e., de-compilation is recognized in the statute. But addressed the scope by using the phrase “doing of any act”, due to definitional ambiguity concerning the scope of the section, it may be prone to vulnerable interpretations. Further, the Copyright Act does not provide for any definition of interoperability nor does it provide any test to assess interoperability, but for the use of the word ‘essential’. The clause is also not clear as to the possible purpose of interoperability.

Would it include “operating interoperability”, for the purpose of targeting interfaces so as to constitute a competitive market replacement of an existing program? Why not de-compilation for purposes moving beyond “operating interoperability”? What about unprotected element in a computer program that may need de-compilation? The author feels that amendment in this direction must be expedited. 2. Redrafting traditional fair use policy keeping in view legislative development in US and EU. 467 3. Fair use of computer programme should also incorporate interoperability with hardware: software need not only to interoperate with other software but also with the hardware. Application programs usually communicate with the operating system which in turn makes the hardware perform the necessary function. Hence, the current reverse engineering prong provided under section 52 (1)(ab) must be logically extended to hardware interoperability. Necessary amendment in this direction must be expedited. 4. Contractual restraints which frustrate the purpose of fair use should not be allowed: Thus amendment in the direction declaring contracts that restrict “fair use” as void must be expedited. This can be achieved by incorporating a clause that “Any contractual provisions contrary to section 52(aa), (ab), (ac), (ad) shall be null and void.” Amendment has to be carried based on the Art 9 of the EU Software Directive. 5. Sec 65 A of the Copyright Amendment Act 2012 needs certain clarification by the legislature to understand its exact scope. Firstly, the clarification must be specific in relation as to whether access control TMs or certain specific activity controlling TMs or both come within the definition of “effective technological measures”. Secondly, the Provision has not defined the term “circumvention” to know the ambit of the activities which will come within its purview. The purpose is to make clarity as to when will liability under the provision be attracted. The provision has left to the judiciary to decide as to which are the activities that will be considered as circumvention for the purpose of attaching liability under the provision. This must also be legislatively clarified and to this effect the provision has to be amended. 6. Establish or strengthen competition laws in order to control, inter alia, possible abuses emerging from the acquisition and exercise of IPRs. 468
allowed by the TRIPS Agreement to determine the grounds for compulsory licenses to remedy anticompetitive practices relating to IPRs considers in particular the granting of compulsory licence in cases of refusal to deal. Also to prevent the formation of cartells. 8. Conceptualise and apply the essential facilities doctrine as required to address situations of control of essential technologies taking into account of relevant market conditions and public needs.

9. Compulsory Licensing as the Fairest Solution: Compulsory licensing should be accepted as a tool for mandatory disclosure by adopting the doctrine of copyright misuse and essential facility evolved by the judiciary in US and EU. Recognizing the compulsory license approach as followed in Microsoft case. 10. Amendment to Competition Act 2002: Sec 3 (5) should go on with a rider to the effect that however, in the exercise of those rights, they would be subject to the relevant provision of the Competition Act. As the section stands, anti-competitive practices in licensing technology would fall under section 3(1), as subsection (3) and (4) deal with specific groups. By interpreting section 3 (4) (d) which deals with “refusal to deal” states that any enterprise enter in to any agreement restrict the person to whom the license was granted from doing any act to understand the necessary interface is considered to be anti competitive and in contravention of section 3 of the Act.

References:

- G Weiner, ‘Reverse Engineering as a Method of Achieving Compatibility in the Computer Industry’ (1997) 6 University of Baltimore Property Law Journal 1.See for example, s. 1 Canadian Copyright Act which defines a computer program as ‘a set of instructions or statements, expressed, fixed, embodied or stored in any manner, that is to be used directly or indirectly in a computer in order to bring about a specific result’.
• Rappaport, I ‘EC Threatens Software Protection’, San Francisco Recorder, Feb 22, 1990, at 6 (Mr Rappaport was acting as Apple’s Intellectual Property Counsel).
• Atari Games Corp v Nintendo of America Inc 975 F.2d 832 (Fed. Cir. 1992).
• Digital Millenium Copyright Act s. 1201(f)(4).