VOLUME 3 | ISSUE 4

INTERNATIONAL JOURNAL OF ADVANCED LEGAL RESEARCH

IDEA EXPRESSION DICHOTIOMY: CHALLENGES TO COMPUTER PROGRAMS

- Harsha Pathak & Anindya Sircar[1]

ABSTRACT

The principle of idea-expression dichotomy is a crucial legal concept that differentiates between safeguarding intangible concepts and the manifestation of those concepts in a concrete format. Within the realm of computer programming, this principle poses notable obstacles in ascertaining the degree of intellectual property entitlements and the breadth of legal safeguarding. The present study delves into the complexities and difficulties entailed in the concept-expression dichotomy within the domain of computer programmes

The application of the idea-expression dichotomy is uniquely challenging when dealing with the abstract nature of computer programmes. Computer programmes are comprised of fundamental concepts, algorithms, and logical frameworks that enable their operation. Conversely, these ideas are also exhibited through a distinct code or programming language that serves as a representation of said concepts. The existence of this duality prompts inquiries as to whether safeguarding should be extended to the fundamental concepts, the precise programming instructions, or both.

The present study investigates the legal frameworks and case law pertaining to the dichotomy between idea and expression in diverse jurisdictions, encompassing the United States, the European Union, and other prominent jurisdictions. The text scrutinises pivotal legal cases and their ramifications for safeguarding computer programmes. Furthermore, the present study delves into the difficulties entailed in demarcating the limits between concepts and articulations, particularly in light of the swift advancement of technology and the growing intricacy of computer software.

This paper explores the ramifications of the dichotomy between idea and expression on the innovation and competition of software. Achieving an optimal equilibrium between

[1] LLM Student, NALSAR University of Law, Hyderabad & DPIIT IPR Chair Professor, NALSAR University of Law, Hyderabad Respectively

safeguarding intellectual property rights and promoting innovation is a multifaceted undertaking. This study examines the potential hindrance to software developers' ability to innovate and build upon existing ideas due to a strict adherence to the idea-expression dichotomy, which may impede technological advancement. This text delves into alternative methodologies, such as fair use and interoperability exceptions, that strive to achieve equilibrium between the concerns of computer programme creators and users.

Keywords: Idea Expression Dichotomy, Computer Programs, Graphic User Interface, Abstraction and Functionality, Reverse Engineering, Source Code, Object Code, Artificial Intelligence (AI).

INTRODUCTION

I was afforded the chance to engage in the study of Intellectual Property Rights through an elective course that was made available to me. I developed a strong interest in copyright laws, which encompass a range of creative aspects and theories, including the Idea-Expression Dichotomy. I have encountered a highly captivating issue that confronts one of the most significant and thriving sectors. The subject matter pertains to the software industry, specifically the issue of copyright pertaining to diverse computer programmes or software. The Idea Expression Dichotomy is a crucial element in achieving the objective of copyright law. The fundamental objective of copyright law is to foster creativity and provide motivation to artists, thereby stimulating greater artistic expression.[2] As per the principle of the Idea-Expression Dichotomy, copyright protection is afforded solely to the expression of ideas, rather than the ideas themselves.[3] It is established that ideas are not subject to copyright protection. The proposition suggests that ideas are not subject to copyright protection, as doing so could impede innovation and ultimately harm society.[4]

The development of computer programmes has enabled individuals to attain billionaire status, as witnessed by the global community. In the absence of intellectual property (IP) protection, would the commercial exploitation of these programmes have yielded significant financial gains

[2] Diva Rai, *Legal analysis of theory of idea-expression dichotomy*, IPLEADERS (2021), https://blog.ipleaders.in/legal-analysis-of-theory-of-idea-expression-dichotomy/ (last visited Jun 7, 2023).

[3] Idea and Expression Dichotomy, https://www.lawteacher.net/free-law-essays/copyright-law/idea-and-expression-dichotomy.php (last visited Jun 7, 2023).

[4] Dipanwita Chatterjee, *Kanoon Gurus | Legal News | Latest Legal & Business News, Updates, Short Summary | Know What is the concept of Idea-Expression Dichotomy ?*, (2021), https://kanoongurus.com/blog/idea-expression-dichotomy/ (last visited Jun 7, 2023).

amounting to billions of dollars? Intellectual property law safeguards computer programmes that are created through expertise and ingenuity, and embody novel concepts aimed at benefiting humanity. Is it justifiable for an individual to replicate an idea and benefit from the efforts of another individual? The aforementioned inquiries served as the impetus for my investigation into a subject matter of this nature.

The safeguarding of software applications presents a noteworthy contradiction within the realm of copyright legislation.[5] Despite the recognition of computer programmes as literary works in both European and international legal frameworks, it remains noteworthy that a computer programme designed to offer travel packages is afforded the same level and length of protection as the most enthralling literary masterpieces of our time. It is noteworthy to acknowledge that computer programmes, being functional works designed to enable machines to perform specific tasks, possess a commercial essence that is not encompassed within the ambit of the conferred protection.[6]

The present study aims to examine the principle of dichotomy between idea and expression and its implementation in the context of copyright disputes related to software or programming infringement. The inquiry pertains to the degree to which the legitimacy of copyright law jurisprudence is justified with regards to the access and utilisation of ideas or information contained within an author's work. The article concludes by examining the public policy implications that underlie the principle of idea-expression dichotomy and its applicability to the software industry.

BRIEF HISTORY OF COMPUTER PROGRAMS

Computer programmes play a crucial role in contemporary technology, facilitating the operation and mechanisation of innumerable tasks. The development of computer programmes can be traced back to the early stages of computing. This concise historical account offers an overview of their evolution, beginning with the earliest programming endeavors and culminating in the advanced software systems that are integral to contemporary society.

The origins of the programmable machine can be traced back to the early 19th century. Charles Babbage, a renowned English mathematician, and inventor conceived the Analytical Engine in

[5] Idea-Expression Dichotomy - JusIP Law Firm, (2021), https://www.jusip.in/idea-expression-dichotomy/ (last visited Jun 7, 2023).

[6] Himanshu-Sinha-.pdf, https://iprlawindia.org/wp-content/uploads/2021/03/Himanshu-Sinha-.pdf (last visited Jun 7, 2023).

1837. [7]This innovative mechanical apparatus was designed to execute computations using punched cards. Even though Babbage's concepts were not entirely implemented during his lifetime, they established the groundwork for forthcoming computing devices.

The advent of electronic computers in the mid-20th century brought about a revolutionary transformation in the field of computing. During the 1940s, a group of trailblazers including Konrad Zuse, John Atanasoff, and Alan Turing constructed the initial electronic computers, which were colossal machines occupying entire rooms. The initial computer systems utilised vacuum tubes and punched cards as the primary means of input and output.[8]

During the early stages of computing, programming required the manual rewiring of the machine, a laborious and susceptible-to-errors task. The pivotal moment occurred with the emergence of assembly languages and high-level programming languages. During the mid-1950s, Grace Hopper created the initial compiler that facilitated the translation of human-readable instructions into machine code. This development represented a noteworthy advancement in the progression of computer programming, rendering it more attainable and effective.[9]

During the 1960s and 1970s, there were notable progressions in the development of programming languages. Fortran, which stands for Formula Translation, was created by IBM and gained widespread adoption as a high-level programming language, particularly in the fields of science and engineering. The Common Business-Oriented Language (COBOL) was a significant programming language that enabled the creation of software systems with a focus on business applications.[10]

During the 1970s, the advent of microprocessors and personal computers facilitated the democratisation of computing power. Programming languages such as BASIC (Beginner's All-purpose Symbolic Instruction Code) and Pascal were developed with the aim of streamlining programming for these novel platforms. Furthermore, the emergence of operating systems, such as UNIX, facilitated a uniform platform for executing and administering computer applications.[11]

[7] Bricsys, *Computer Programing a Brief History | Bricsys Blog*, BRICSYS-WEBSITE, https://www.bricsys.com/blog/computer-programing-a-brief-history (last visited Jun 7, 2023).

[8] *Id.*

[9] A History of Computer Programming Languages, ONLINE COLLEGE PLAN (2018), https://www.onlinecollegeplan.com/computer-programming-languages/ (last visited Jun 7, 2023).

[10] History of computers: A brief timeline | Live Science, https://www.livescience.com/20718-computer-history.html (last visited Jun 7, 2023).

[11] Computer History for 1970, https://www.computerhope.com/history/1970.htm (last visited Jun 7, 2023).

During the 1980s and 1990s, there was a notable expansion in the software sector. The advent of graphical user interfaces (GUIs) and object-oriented programming (OOP) languages, such as C++ and Java, brought about a significant shift in the development and utilisation of software. The aforementioned technological progressions facilitated the production of software applications that are more accessible to users and stimulated the growth of the internet and the global network of interconnected computer systems known as the World Wide Web.[12]

The advent of the 21st century has witnessed a swift escalation in the intricacy and efficacy of computer software. The prominence of open-source software, as demonstrated by the Linux operating system and the Apache web server, has facilitated collaboration and innovation. Furthermore, the widespread adoption of mobile devices and the emergence of cloud computing have created novel opportunities for the development and distribution of software.[13]

To ascertain the process of claiming copyright on a computer programme, it is imperative to have a clear understanding of the definition of a computer programme. In layman's terms, a computer programme refers to a collection of directives that are utilised to execute tasks.[14] The process involves an individual transmitting a set of instructions, such as data, algorithm, data structure, among others, to a computing device, which subsequently executes a task based on the provided information. It can be posited that computer programmes are a form of language that is comprehensible to computing systems. Assuming that a specific task is to be performed by an individual, it is necessary to provide instructions in a language that is comprehensible to said individual.

The set of instructions can be conveyed through various syntactical structures, yet the task to be executed by the individual remains constant. Similarly, in the realm of computer programmes, while the output produced by a computer may remain constant, the specific sequence of instructions provided to the computer can vary.[15] The matter of copyright arises during the act of providing directives to the computer.

As per the WIPO Copyright Treaty of 1996, the World Intellectual Property Organisation has affirmed that computer programmes are classified as literary works. As per the provisions

[12] Computers in the 1990s - History-Computer, https://history-computer.com/computers-in-the-1990s/ (last visited Jun 7, 2023).

[13] The computer for the 21st century: present security & privacy challenges | Journal of Internet Services and Applications | Full Text, https://jisajournal.springeropen.com/articles/10.1186/s13174-018-0095-2 (last visited Jun 7, 2023).

[14] Computer program | Definition & Facts | Britannica, https://www.britannica.com/technology/computer-program (last visited Jun 7, 2023).

[15] Webopedia Staff, *What is a Program? (Definition & Meaning)*, WEBOPEDIA (1996), https://www.webopedia.com/definitions/program/ (last visited Jun 7, 2023).

outlined in Article 4 of the World Copyright Treaty (WCT), the term "computer" shall refer to any device or apparatus that is capable of processing digital data and executing instructions to produce a desired result.[16]

According to Article 2 of the Berne Convention, programmes are safeguarded as literary works.[17] Article 4 of the copyright law stipulates that computer programmes are eligible for copyright protection, regardless of their mode or form of expression.[18] This provision implies that programmers are entitled to the same level of copyright protection as literary authors. According to Article 7 of the World Trade Organization's Agreement on Trade-Related Aspects of Intellectual Property Rights (WTO TRIPS), computer software programmers are prohibited from exercising their exclusive right to authorise commercial rental of the original or copied software to the public if the programme is not the primary object of the rental.[19]

Source code and Object code and its importance in Copyrightability of a Computer Programs

The concepts of source code and object code are fundamental in computer programming. Source code refers to the human-readable instructions that a programmer writes in a high-level programming language. Object code, on the other hand, is the machine-readable version of the source code that is generated by a compiler or an assembler.[20] These two types of code are essential in the software development process and play a crucial role in the execution of computer programmes.

The term "source code" pertains to the primary code that is created by a programmer in a high-level programming language, such as Java, Python, or C++. This code is legible to humans. The program's structure, which is crafted and modified by programmers prior to its compilation or interpretation into machine code, is commonly referred to as its source code. The copyrightability of source code is generally acknowledged due to its representation of the

[16] WIPO COPYRIGHT TREATY (WCT) (1996): WITH THE AGREED STATEMENTS OF THE DIPLOMATIC CONFERENCE THAT ADOPTED THE TREATY AND THE PROVISIONS OF THE BERNE CONVENTION (1971) REFERRED TO IN THE TREATY, (1997).

[17] WIPO Lex, https://www.wipo.int/wipolex/en/text/283698 (last visited Jun 7, 2023).

[18] WIPO COPYRIGHT TREATY (WCT) (1996), *supra* note 15.

[19] trips_art7_oth.pdf, https://www.wto.org/english/res_e/publications_e/ai17_e/trips_art7_oth.pdf (last visited Jun 7, 2023).

[20] Difference between Source Code and Object Code - GeeksforGeeks, https://www.geeksforgeeks.org/difference-between-source-code-and-object-code/ (last visited Jun 7, 2023).

program's concepts and ideas.[21]

In contrast, object code refers to the program's machine-readable iteration that is produced by the compiler or interpreter. The process of generating machine code involves the conversion of the original source code into a binary format that is capable of being interpreted and executed directly by the computer. The functional representation of a programme, which is object code, is generally not deemed eligible for copyright protection as it does not constitute an expression of the underlying ideas and concepts.[22]

IDEA EXPRESSION DICHOTOMY

The doctrine of Idea Expression Dichotomy serves as a means of discerning between the conceptualization of an idea and the articulation of said idea.[23] Dichotomy refers to the act of making a clear distinction or division between two contrasting or opposing concepts or entities. The legal entitlement known as copyright is bestowed upon the author or creator of a work. Copyright law safeguards the intellectual property of the originator and confers upon them the sole authority to exercise control over their creation. However, copyright does not preclude other creators from generating works based on the same concept, provided that the manifestation is distinct. Copyright protection is granted to a diverse range of creative works, including but not limited to books, music, paintings, sculptures, films, computer programmes, databases, advertisements, maps, and technical drawings.[24]

In essence, copyright protection is limited to tangible expressions of ideas that have been fixed in a certain form. The concept in question does not possess an inherent copyright. The concept can be comprehended through a straightforward illustration. Consider a scenario where two individuals conceive the notion of composing a brief narrative, such as one of the tales from Sherlock Holmes. If one of the writers, namely Sir Arthur Conan Doyle, commits the story to writing and secures copyright protection, the other writer cannot assert copyright solely based on having conceived the same idea. This demonstrates that legal protection is only conferred upon an idea if it is expressed in a tangible form.

[21] What is Source Code in Programming and How Does It Work?, https://www.techtarget.com/searchapparchitecture/definition/source-code (last visited Jun 7, 2023).

[22] What is object code? – TechTarget Definition, WHATIS.COM, https://www.techtarget.com/whatis/definition/object-code (last visited Jun 7, 2023).ob

[23] Baker v. Selden | Case Brief for Law Students | Casebriefs, https://www.casebriefs.com/blog/law/intellectual-property-law/intellectual-property-keyed-to-merges/copyright-law/baker-v-selden/ (last visited Jun 7, 2023).

[24] Yogesh A. Pai, *Copyright Protection for Computer Programs: Walking on One Leg?*, 48 J. INDIAN LAW INST. 359 (2006), https://www.jstor.org/stable/43952047 (last visited June 6, 2023).

The rationale behind the absence of copyright protection for ideas, but rather solely for their manifestation, warrants examination. One primary rationale is to safeguard the unimpeded dissemination of concepts and thoughts. The implementation of intellectual property rights may have a negative impact on creativity within society, as it could potentially restrict the unrestricted exchange of ideas. The principle of granting freedom to reproduce ideas is a fundamental tenet of copyright law, underscoring its critical importance.[25]

The Doctrine of Merger refers to a situation where the interdependence of ideas and expression is so significant that they cannot be distinguished from one another, resulting in the loss of protection for both. According to the legal precedent set by "Jeffrey's v. Boosey,"[26] abstract ideas are not afforded protection. However, the tangible manifestation of those ideas, in the form of material expression, is eligible for legal protection. The court in the case of "Donoghue vs Allied Newspaper Ltd"[27] made an observation that the act of expressing an idea or a story does not confer copyright ownership upon the expresser, as it does not entail any skill in expression. The individual who employs their expertise, erudition, and ingenuity to compose a narrative is the individual who may hold the copyright. The well-known legal case of "RG Anand v. Deluxe Films"[28] in India similarly involved a dispute of this nature. The individual bringing forth the legal action had authored and produced a theatrical production entitled "Hum Hindustani." The plaintiff initiated legal action for copyright infringement against the defendant, contending that the defendant's motion picture titled "New Delhi" was a reproduction of the plaintiff's own work. Both the Trial Court and the High Court have determined that there was no violation of copyright. Subsequently, the plaintiff filed a Special Leave Petition with the Supreme Court for further consideration. According to the Supreme Court, while the concepts may have been comparable, the mode of their articulation was distinct enough to preclude any violation of copyright. The court's verdict was that there was no infringement of the plaintiff's work.

[25] Jon O. Newman, *New Lyrics for an Old Melody: The Idea/Expression Dichotomy in the Computer Age The Sixth Annual Herbert Trenzor Distinguished Lecture in Intellectual Property*, 17 CARDOZO ARTS ENTERTAIN. LAW J. 691 (1999), https://heinonline.org/HOL/P?h=hein.journals/caelj17&i=703 (last visited June 6, 2023).

[26] Jefferys v. Boosey (United Kingdom), Privy Council, Judgment, Law, casemine.com, HTTPS://WWW.CASEMINE.COM, https://www.casemine.com/judgement/in/5779f1d4e561096c9312f615 (last visited Jun 7, 2023).

[27] Donoghue Vs Allied Newspaper Ltd - ; (1937) 3 ALL ER 503 Facts: In 1931, the News of the World paper - Studocu, https://www.studocu.com/in/document/university-of-delhi/bachelor-of-law-du-llb-entrance-exam-notes/donoghue-vs-allied-newspaper-ltd/49736969 (last visited Jun 7, 2023).

[28] R.G Anand vs M/S. Delux Films & Ors on 18 August, 1978, https://indiankanoon.org/doc/1734007/ (last visited Jun 7, 2023).

COMPUTER PROGRAMS: PROTECTION AND ENFORCEMENT

Numerous instances, such as the Whelan case and the recent dispute between Google and Oracle, have raised the question of the copyrightability of computer software. In such instances, the judiciary has encountered challenges in ascertaining the protectable components of computer programmes.

The legal precedent that established the concept of idea-expression dichotomy in computer software was the Whelan v Jaslow[29] case. The case involved Rand Jaslow's intention to develop software for Jaslow Dental Labs that would facilitate billing, accounting, and customer management. The individual engaged the services of the Strohl system to facilitate the development of the software. The software was developed by Elaine Whelan in 1979, who also held a 50% ownership stake in the product. The software, which was christened Dentalab by Strohl, was retained under his ownership. However, it was also made available for licensing to other firms, with Jaslow receiving a 10% commission. Subsequently, in 1979, Whelen departed from Strohl and established a new enterprise, acquiring the software rights.

Jaslow established Dentlab in 1982, a company that produced comparable software to Dentalab. The product exhibited novel functionalities that were absent in Dentalab. Subsequently, Jaslow initiated legal proceedings claiming that Dentlab had violated the copyright of Whelans in the Dentalab software. In this case, the Court established a criterion that differentiated the components that were eligible for legal safeguard and those that were not. The notion posited that any element that contributes to the operation of the software is intrinsic to the software, and that such an intrinsic element constitutes the concept. The extraneous components of the software are those that do not contribute to its functionality, and instead serve as a means of conveying ideas. The court determined that the new iteration bears a significant resemblance to Dentalab, and consequently, the latter was awarded copyright protection. A highly rigid interpretation of the expression of ideas was implemented, leading to an impediment to the advancement of novel concepts.

The legal case of Apple Computer Inc versus Microsoft Corp[30] involved the introduction of Apple's initial commercially available computer in 1983, which featured a Graphic User Interface modelled after Xerox. In 1985, Microsoft introduced Windows 1.0 and entered into a

[29] Whelan Associates, Inc. v. Jaslow Dental Laboratory, Inc., (2023), https://h2o.law.harvard.edu/cases/5004 (last visited May 31, 2023).

[30] Apple Comput., Inc. v. Microsoft Corp. | Case Brief for Law School | LexisNexis, COMMUNITY (2023), https://www.lexisnexis.com/community/casebrief/p/casebrief-apple-comput-inc-v-microsoft-corp (last visited May 31, 2023).

licensing agreement with Apple for specific graphical user interface (GUI) features that were originally derived from Xerox. Conversely, Apple has permitted the utilisation of its products for commercial purposes on its computers. Apple initiated legal proceedings against Windows 2.0, alleging that the graphical user interface (GUI) of Windows had been derived from Apple's proprietary GUI. In support of its claim, Apple submitted a catalogue of 189 features to the court that bore resemblance to those of the Macintosh. The court granted a license for only 179 of these features.

The court of the district rendered a verdict in favour of Apple. Microsoft filed an appeal to the Appellate court, which subsequently determined that 90% of the features were deemed to be within the acceptable threshold of agreement. The remaining items were deemed to be of a generic nature, and therefore, copyright protection could not be granted to them. Except for a small number of features, Apple's claim was unsuccessful. The primary limitation of the case was the absence of Xerox, whose graphical user interface served as the foundation for the interfaces of both Apple and Microsoft, as a party to the litigation.

In the subsequent legal matter involving Lotus Development Corporation. In the case of Corp. vs Borland Inc,[31] the court examined the scope of software copyright protection. The plaintiff in this case is Lotus Corporation, which has a product available in the market under the name Lotus 1-2-3. The Lotus 1-2-3 software package was comprised of a total of 456 distinct commands. Several commands were utilised, including print, paste, copy, cut, and others. Subsequently, Borland introduced a spreadsheet application which incorporated the complete menu structure of Lotus 1-2-3, albeit without replicating the underlying code utilised by Lotus. The Lotus company initiated legal proceedings in a district court on the grounds of copyright infringement. The court subsequently ruled in favour of Lotus, thereby granting them a favourable judgement. Borland lodged an appeal and asserted that he solely utilised the process or procedure, which falls beyond the purview of safeguarding against copyright infringement. According to the appellate court, the hierarchy menu constitutes a "method of operation" or a process that does not fall under the protection of copyright law. The court drew a comparison between the VCR and the hierarchical menu. As with the control panel of a VCR, the commands within a hierarchical menu are crucial for the operation of a programme.

[31] Lotus Dev. Corp. v. Borland Int’l | Case Brief for Law School | LexisNexis, (2023), https://www.lexisnexis.com/community/casebrief/p/casebrief-lotus-dev-corp-v-borland-int-l (last visited May 31, 2023).

The case of Computer Associates International versus Altai[32] involved the development of a software programme called OSCAR 3.4 by a former employee of Computer Associates for Altai. The source code of the programme was found to contain 30% copied content from ADAPTER. Subsequently, ALTAI proceeded to revise the software, specifically OSCAR 3.5, and asserted that it was not a replication of ADAPTOR. This was due to the fact that the programmers who were involved in the development of OSCAR 3.5 had no involvement in the creation of OSCAR 3.4. The district court determined that infringement was absent in OSCAR 3.5, while it was present solely in OSCAR 3.4. The plaintiff lodged an appeal on the grounds of copyright infringement pertaining to OSCAR 3.5. A legal standard was developed by the court to ascertain the presence of copyright infringement. The assessment was formally referred to as the Abstraction Filtration Comparison test. As per the results of this assessment, the initial step would entail deconstructing the programmes that have been purportedly violated into their elemental components. Subsequently, it is necessary to scrutinise all components vis-à-vis concepts that fall within the purview of the public domain, wherein replication is permissible. This will enable the court to exclude all the non-protectable content. The final stage of the court's process will involve a comparison between the material and the architecture of the programme that is being accused of infringement. The outcome of this comparison will ascertain whether the elements of the programmes in question that are eligible for protection exhibit significant similarity, thereby leading to a conclusion of infringement.

Upon implementing this procedure, the court arrived at the determination that there existed no violation of copyright on OSCAR 3.5.

The legal dispute between Google LLC and Oracle America Inc.[33] has gained significant notoriety and is commonly referred to as the "case of the decade". In 1990, Sun Microsoft created a novel programming language known as JAVA. After a decade and a half, Google acquired Android from a fresh developer and incorporated JAVA's API during the development of its mobile operating system for various devices, including smartphones. In the year 2010, Oracle acquired Sun Microsystems, citing allegations of copyright infringement. The Supreme Court employed a four-factor assessment to determine the applicability of fair use. The fundamental factors that determine the permissibility of using copyrighted material are the nature of the work, the purpose and character of the use, the quantity of the copyrighted

[32] Computer Associates v. Altai (BitLaw), (2023), https://www.bitlaw.com/source/cases/copyright/altai.html (last visited May 31, 2023).

[33] 18-956 Google LLC v. Oracle America, Inc. (04/05/2021), (2021).

material, and the impact on the market for the copyrighted material. According to the court's ruling, Google utilised Java programming language primarily due to its familiarity among developers, rather than its aesthetic or innovative qualities. Therefore, the court contended that Google's utilisation of JAVA fell within the scope of the fair use doctrine due to these justifications.

COMPUTER PROGRAMS-CHALLENGES FOR COPYRIGHT

1. Abstraction and Functionality:The interplay between abstraction and functionality can pose a challenge in the context of computer programmes, particularly with regards to the dichotomy of idea expression. The amalgamation of concepts and functional components in a computer programme poses a challenge in distinguishing the safeguarded expression from the fundamental ideas.[34]

   One of the challenges stems from the inherent nature of computer programmes being designed to execute specific functions or tasks. The functions are deemed as fundamental components warranting safeguarding, as they stem from innovative notions and concepts. The articulation and execution of these concepts necessitate the composition of code, which serves as a precise manifestation of said concepts. Delineating the distinction between the concept and manifestation of a programme can prove to be a formidable task, particularly when the operational aspects of the programme are inextricably linked with its implementation particulars.[35] Ascertaining the appropriate level of protection to be afforded can prove to be a formidable task when addressing intangible notions.
2. Reverse Engineering: Reverse engineering refers to the systematic examination of a product, system, or programme with the aim of comprehending its design, structure, functionality, or implementation particulars. Reverse engineering in the realm of computer programmes refers to the process of analysing a compiled or executed programme to obtain insights into its fundamental concepts, algorithms, or data structures. Reverse engineering serves multiple functions, including comprehension of programme functionality and the development of interoperable systems. However, it presents difficulties concerning the dichotomy of idea expression and the safeguarding of intellectual property. In the process of reverse engineering, distinguishing the fundamental concepts or principles from their manifestation in the programme can pose a challenge. The implementation of a programme

[34] Whelan Associates, Inc. v. Jaslow Dental Laboratory, Inc., *supra* note 28.

[35] Lotus Dev. Corp. v. Borland Int'l | Case Brief for Law School | LexisNexis, *supra* note 30.

may be closely linked to its underlying concepts or functionality, including factors such as code syntax, data structures, and algorithmic decisions. The act of reverse engineering may prompt inquiries into the potential infringement upon the copyright protection of a program's expression.

Reverse engineering poses a challenge due to its legal implications.[36] The legality of reverse engineering is subject to variation across different jurisdictions and is contingent upon several factors, including but not limited to the purpose of the reverse engineering activity, the characteristics of the programme in question, and the relevant legal framework.[37] Reverse engineering may be permissible in certain instances, such as for the purposes of promoting interoperability or conducting security testing. In certain instances, there may exist prohibitions or limitations on the use of software to safeguard the intellectual property rights of its developers.

3. Merger Doctrine: The merger doctrine poses a significant challenge in the domain of computer programmes, owing to the inherent complexities of programming languages, algorithms, and coding practices. In certain cases, particular functional requirements or algorithms may mandate a specific expression or code sequence, thereby restricting the available alternatives.[38] The restricted scope of linguistic expressions may give rise to the amalgamation of concepts and their articulation, which could potentially lead to diminished or nullified copyright safeguarding.

   An instance of the merger doctrine's implementation in computer programmes pertains to safeguarding frequently employed algorithms or programming methodologies. Algorithms that pertain to sorting or searching are fundamental and extensively utilised in the field of computer programming. The functionality of these algorithms is contingent upon their implementation details, which may have a restricted range of expression.[39] The merger doctrine has the potential to classify the expression as indivisible from the underlying idea, thereby reducing the extent of copyright protection that can be obtained.

4. Functional Compatibility: The notion of interoperability and functional compatibility poses considerable obstacles to the dichotomy between idea expression and computer programmes. Interoperability pertains to the capacity of disparate systems or components

[36] Sony Computer Entertainment, Inc. v. Connectix Corp. on JSTOR, https://www.jstor.org/stable/24119912 (last visited Jun 7, 2023).

[37] Sega Enters. v. Accolade, Inc. | Case Brief for Law School | LexisNexis, https://www.lexisnexis.com/community/casebrief/p/casebrief-sega-enters-v-accolade-inc (last visited Jun 7, 2023).

[38] Computer Associates v. Altai (BitLaw), *supra* note 31.

[39] Sega Enters. v. Accolade, Inc. | Case Brief for Law School | LexisNexis, *supra* note 36.

to function in unison and facilitate the seamless exchange of information. On the other hand, functional compatibility pertains to the ability of software programmes to furnish comparable functionalities or services. Both concepts are essential in fostering innovation, competition, and enhancing user convenience.[40] Nevertheless, they may pose a challenge in the copyright protection of computer programmes by complicating the differentiation between safeguarded expression and unprotected ideas.

Achieving interoperability and functional compatibility frequently necessitates a certain level of uniformity or compatibility in the configuration, layout, or procedures employed by software applications. The maintenance of uniformity guarantees efficient comprehension and communication between disparate systems. However, when it comes to copyright protection, the need for interoperability and functional compatibility can clash with the idea expression dichotomy.

The legal dispute between SAS Institute Inc. and World Programming Ltd.[41] in 2012, as heard by the European Court of Justice (ECJ), offers valuable insights into the difficulties associated with achieving interoperability and functional compatibility. The present case involves a claim made by SAS Institute, a software enterprise, asserting that World Programming's software, which replicated the functionality of SAS software, violated its copyright. The European Court of Justice (ECJ) underscored the significance of interoperability and functional compatibility in its verdict that World Programming did not violate copyright laws by producing a programme that possessed functional similarity to SAS software and had the ability to read and write files in the same format.

Moreover, the promotion of interoperability and functional compatibility is significantly influenced by open standards and industry norms. Open standards refer to publicly available and widely adopted specifications, protocols, or file formats that facilitate seamless interaction and data exchange among various software programmes. The utilisation of open standards may pose difficulties in safeguarding copyright as they frequently embody prevalent and widely adopted concepts or functionalities.

GLOBAL REACTIONS

The response of the international community to these challenges has been diverse,

[40] 18-956 Google LLC v. Oracle America, Inc. (04/05/2021), *supra* note 32.

[41] Tatiana Synodinou, *Decrypting the code: CJEU SAS vs. World Programming*, KLUWER COPYRIGHT BLOG (2012), https://copyrightblog.kluweriplaw.com/2012/05/07/decrypting-the-code-cjeu-sas-vs-world-programming/ (last visited June 6, 2023).

encompassing legal modifications, technological progress, and ongoing deliberations regarding the equilibrium between innovation and safeguarding.

One of the primary obstacles pertains to the creation and execution of computational algorithms capable of distinguishing and upholding the distinction between ideas and their expression. With the advancement of artificial intelligence (AI), there arises a growing necessity for machines to comprehend the subtle distinctions between concepts and their particular articulations. The development of algorithms and models that can effectively analyse and interpret the nuances of human creativity, while simultaneously avoiding any potential infringement on copyrighted material, is imperative.[42]

The challenges at hand have prompted a worldwide response, which entails continuous deliberations among policymakers, legal scholars, and technology industry participants. In the digital age, it is imperative to establish a comprehensive framework that strikes a balance between protecting intellectual property rights and fostering innovation. This necessitates collaborative efforts. Enhancing international collaboration and harmonising legal frameworks can promote a more uniform strategy, guaranteeing lucidity for software applications and their creators in navigating the dichotomy between idea and expression.[43]

PRESENT LEGAL FRAMEWORK

The legal frameworks of various jurisdictions, such as the United States (US), the European Union (EU), and India, pose challenges for computer programmes with regards to the idea-expression dichotomy. The protection and scope of copyright for computer programmes are shaped differently by various jurisdictions.

Computer programmes are safeguarded under the copyright law in the United States as literary works. The Copyright Act acknowledges that the safeguarding of copyright pertains to the manifestation of concepts, rather than the concepts themselves. This means that the specific code and structure of a computer program are eligible for protection, while the underlying ideas or functionalities are not. The legal system in the United States also encompasses the fair use doctrine, which permits specific and restricted utilisation of copyrighted material without obtaining permission, such as for the purposes of criticism, commentary, or education.[44]

[42] Artificial intelligence and copyright, https://www.wipo.int/wipo_magazine/en/2017/05/article_0003.html (last visited Jun 7, 2023).

[43] The computer for the 21st century: present security & privacy challenges | Journal of Internet Services and Applications | Full Text, *supra* note 12.

[44] circ61.pdf, https://www.copyright.gov/circs/circ61.pdf (last visited Jun 7, 2023).

The Software Directive and the Copyright Directive provide protection for computer programmes within the European Union. According to the Software Directive, the manifestation of a computer programme in any format, such as its source code and object code, is eligible for copyright protection. The European Union's framework places significant emphasis on the crucial role of interoperability and competition, which entails the lawful practice of reverse engineering computer programmes to attain compatibility. Furthermore, the copyright framework of the European Union acknowledges the notion of rights exhaustion. This implies that the copyright owner's authority over the distribution of a computer programme is depleted once a legitimate copy is sold within the EU.[45]

As per the Copyright Act in India, computer programmes are safeguarded as literary works. The legal framework in India prioritises safeguarding the manifestation of concepts as opposed to the concepts per se. Copyright protection can be granted to computer programmes if they meet the criteria of being both original and fixed in a tangible medium. The Indian copyright legislation encompasses provisions for fair dealing, which permit uses of copyrighted material without violating the copyright owner's rights.[46]

In order to tackle these obstacles, there are ongoing deliberations, legal rulings, and legislative modifications occurring in every jurisdiction. The interpretation and application of the idea-expression dichotomy in specific cases is a critical function of courts, while policymakers and legislators endeavor to achieve a balance between safeguarding originality and fostering innovation.

FUTURE OF COMPUTER PROGRM PROTECTION

The future of safeguarding computer programmes may be facilitated by the establishment of unambiguous criteria and benchmarks to demarcate the demarcation between concepts and their articulation. Collaborative efforts between courts and policymakers can be employed to establish standardised and impartial criteria and assessments that can be consistently implemented across various jurisdictions. Enhancing the lucidity of copyright protection for computer programmes would offer improved comprehension to creators, users, and developers concerning its extent.

Moreover, the progressions in the field of artificial intelligence (AI) and machine learning are

[45] EUR-Lex - mi0016 - EN - EUR-Lex, https://eur-lex.europa.eu/EN/legal-content/summary/computer-programs-legal-protection.html (last visited Jun 7, 2023).

[46] Indian Copyright Software, https://www.legalservicesindia.com/article/855/Indian-Copyright-Software.html (last visited Jun 7, 2023).

anticipated to influence safeguarding computer programmes. With the increasing sophistication of AI systems, there is a possibility that they may be capable of generating computer programmes or playing a role in the creative process. The emergence of AI-generated programmes has prompted inquiries regarding their ownership and copyrightability. Additionally, there is a need to explore the extent to which AI can aid in navigating the idea-expression dichotomy.[47]

CONCLUSIONS

Upon analysing the cases, it has become evident that distinguishing ideas from expression in computer programmes is increasingly challenging. Despite the courts' diligent efforts and implementation of various assessments, it is evident that the rapid progression of technology will pose significant challenges for the judiciary, potentially resulting in a lag in legal regulations. The dichotomy between the idea and expression presents notable obstacles within the realm of computer programming. The intricacies stem from multiple elements such as abstraction and functionality, reverse engineering, the merger doctrine, user interfaces and APIs, interoperability and functional compatibility, rapid technological advancements, and legal lag. The challenges frequently obscure the demarcation between safeguarded expression and non-safeguarded concepts, thereby posing a challenge in ascertaining the extent of copyright protection applicable to computer programmes. In the domain of computer programmes, the dichotomy between idea and expression encounters several challenges. The intricate issues that stem from abstraction and functionality, reverse engineering, the merger doctrine, user interfaces (UIs) and application programming interfaces (APIs), interoperability and functional compatibility, swift technological progress, and legal lag necessitate meticulous examination and adjustment of legal frameworks.

The prospective outlook for safeguarding computer programmes entails the potential for enhanced lucidity, inventive licencing frameworks, and global collaboration. The establishment of more precise criteria for discerning the boundaries between ideas and expressions, the consideration of the ramifications of programmes generated by artificial intelligence, and the adaptation to the changing requirements of both creators and users will constitute pivotal elements in the configuration of the forthcoming legal structure. Achieving an optimal

[47] The promises and risks of AI in software development | TechTarget, https://www.techtarget.com/searchitoperations/feature/The-promises-and-risks-of-AI-in-software-development (last visited Jun 7, 2023).

equilibrium is crucial for the advancement of safeguarding computer programmes, as it can facilitate sustained ingenuity and safeguard the entitlements and welfare of both producers and consumers in an ever-expanding digital landscape.

Through the resolution of these obstacles, it is possible to cultivate a conducive atmosphere for creativity and originality, while simultaneously upholding the necessary safeguarding of proprietary entitlements in software. In order to address this issue, it is imperative to establish distinct legal frameworks that are tailored to effectively manage this particular phenomenon. Therefore, it is imperative for experts and the World Intellectual Property Organisation (WIPO) to develop novel legal frameworks that can effectively cater to the requirements of the software sector.

*****