NLUA Journal of Intellectual Property Rights ISSN: 2583-8121 (Online)
Volume 2 Issue 1

SOFTWARE INTEROPERABILITY: REVISITING CAMPBELL IN GOOGLE V. ORACLE

Dr. Lucky George[*]

Abstract

Copyright confers an exclusive right to the author as an incentive to create the work and these rights are subject to limitations to promote progress in science and art. Limitation on infringement liability includes first sale doctrine and fair use doctrine. Fair use is the most wide ranging limitation on copyright protection. It is not a relatively narrow rule like limitations but it is encoded as standard and courts develop it on a case to case basis. Though there are recognisable categories of fair use cases, outcomes within each category are unpredictable. The US Supreme Court in the Campbell Case laid down the test of transformative in considering parody as a fair use of original songs resulting in a new expression and not a substitute for the original work. The creation of transformative work is to further the purpose of the copyright law. But the test of transformative should not be misused by allowing the verbatim copy of the original work and apply in the different context without any change in the purpose and character of the original work. Google copied the declaring code verbatim namely the labels of the particular tasks in the API and organizes those tasks, or 'method', into 'packages' and 'classes' stating the user (programmer) has spent considerable time in learning the Java Programming Language and it will be easy for them to write new application programme applying the same language in any new platform. The Supreme Court has given a Judgment in favour of Google allowing the copying as a fair use for the interoperability purpose by revisiting the test of transformative laid down in Campbell case. The copying involves transformation to be considered as the transformative work in that the word 'transform' is used in derivative work over which the original copyright holder retains exclusive control. The word 'transform' results in something new and different from the original or expands its utility which is considered as fair use but copying in any other form without changing the purpose and character is an infringement of derivative rights. This paper critically analyses the Google v. Oracle decision given by the Federal Circuit, Supreme Court Majority and Minority

[*] Associate Professor & Head, Department of IPL, The Tamil Nadu Dr Ambedkar Law University, Chennai.

considering the copying of Java SE software platform in the Google acquired Android, a Startup firm to develop a software platform for mobile devices like smartphones as a fair use. Copying of the declaring code in a new platform of Android is transformative or derivative work where former is considered as a fair use and latter is considered as an infringement of the copyright.

Keywords: Interoperability, Transformative work, Derivative, Application Programme Interface (API), Declaring Code, Method Calls, Implementing Code

1. Introduction

Computer programme is protected as a literary work under the copyright law. It is a useful article which performs a particular task. Computer programme is more informational or functional than creative and it generally favours fair use in allowing the copying of the code to discover the interface or functional element necessary for compatibility and interoperability of the programme. Two Ninth Circuit case in *Sega*[1] and *Sony*[2] allowed the reverse engineering of the programme to discover the functional element required for the interoperability of the independently created programme and interface required to emulate the Sony PlayStation in Apple iMac to enable playing the video game even without the Sony PlayStation console and television. In addition to following *Sega* analysis, the Court held that the Virtual play station is modestly transformative and considered as a fair use. In *Google LLC* v. *Oracle America, Inc.*,[3] the Supreme Court revisited the *Campbell*[4] test of transformative in allowing the copying of the 37 packages of the declaring code of Java SE API in *Google's* Android as a fair use. Google copied the code stating millions of programmers spent considerable time in learning the Java language and used the Java language to write programmes that were able to run on any desktop and laptop using the Java SE Platform. Adopting the same code in Android smartphones, the programmer well versed with Java language can write the application programme for Android, it is user (programmer) interoperable. Dissenting Judgment by Justice Thomas emphasised that copying the code verbatim and structural organisation of the Java SE API is not considered as the transformative work but derivative work and it is an infringement of Copyright under Section 106(2). The paper

[1] *Sega Enterprises Ltd*. v. *Accolade, Inc*., 977 F.2d 1510 (9th Circuit 1993).

[2] *Sony Computer Entertainment Inc.* v. *Connectix Corp.*, 203 F.3d 596 (9th Circuit 2000)

[3] *Google LLC* v. *Oracle America, Inc.*141 S. Ct. 1183 (2021).

[4] *Luther R Campbell* v. *Acuff-Rose Music, Inc.*, 510 US 569 (1994).

critically analyses the case of *Google* v *Oracle*, examining whether the copying of the code is transformative or derivative and to understand the scope of fair use with respect to computer programmes. Further, it examines whether the declaring code naming and the structure is a process or method of operation excluded from copyright protection as laid down in the case of *Lotus Development Corp*. v. *Borland International Inc*.[5]

1.1. Understanding Computer Software

Computer Software is an intangible component of the computer or other hardware that helps direct their operations. Software makes the hardware to work and software is stored in computer memory. Programmers write these instructions to the computer in a programming language, using the syntax of the language like C++ or Python. These instructions are known as Source Code. Source Code cannot instruct the computer in and of itself but it must be converted to a form that a computer can use. A program called a compiler converts the source code to instructions that the computer can execute. This conversion happens by breaking down each source code instructions into a set of computer readable instructions. This converted code that the computer can read or execute is called object code or machine code. It is encoded in binary form, or in zeros and ones. A computer can execute this object code as an application programme in interaction with a computer's operating system. Each programme has to be interoperable/compatible with other programmes to do the particular task or to achieve a particular result.

1.2. Network Externalities in Software

One important feature of the software industry is that each programme has its value if it can operate with other programmes. The demand for one programme depends on the compatibility with other programmes. A market exhibits network effects when "the value that consumers place on a good increases as others use the good".[6]

This can be achieved by sharing interfaces necessary for the compatibility. The computer hardware, operating system, application software and user communicate with each other across 'interfaces'. The system communicates with the user through the 'user interface' which consists of icons on the monitor as well as the keyboard and mouse. The interfaces

[5] *Lotus Development Corp*. v. *Borland International Inc*. 49 F.3d 807 (1st Cir. 1995).

[6] Mark A Lemley and David Mc. Gowan, "Legal Implications of Network Economic Effects", 86(479) *California Law Review* 481 (1998).

between the application programme, operating system and the hardware are internal, and are invisible to users. The underlying interface is kept as a secret and not shared with other programmers. To what extent the interface can be copied to write a programme interoperable with existing programmes, making it easy for the programme developer to write a programme without learning the new code and employing new programmers.

2. Copyright Protection for Software

The object of copyright law is to establish a delicate equilibrium. On the one hand, it must afford protection to the author of the programme as an incentive to create, and, on the other, it must appropriately limit the extent of that protection so as to avoid the effects of monopolistic stagnation in the software industry. The necessary balance between the creative incentive and industrial competition has to be emphasised while conferring protection to any work created out of new technological development. This balance is emphasised in Article 1 Section 8 of the US Constitution *"to promote the progress of Science and Useful Arts, by securing for limited times to author and inventors the exclusive rights to their respective writings and discoveries"*. There are three basic conditions required for obtaining copyright protection. First, there must be a 'work of authorship'[7], that work must be 'original'[8] and the work must be 'fixed[9] in any tangible medium of expression'.[10]

Second, computer programs are protected as a literary work under the copyright law. Computer program is defined as a set of statements or instructions to be used directly or indirectly in a computer to perform a particular task or to achieve a particular result.[11] The definition contains two elements - literal elements and non-literal elements. Literal elements include the programming language in which the programme is written; both the object code and source code are protected as a literary work.[12]

In *Apple Computer Inc.* v. *Franklin Computer Corp.*,[13] *Apple* Computer Inc., sued *Franklin* Computer Corporation for copyright infringement. Franklin had made 'Apple compatible' computers by copying the code for the Apple II operating system programs. This enabled Franklin's consumers to use peripheral equipment as well as the software developed for the Apple II Computer. Franklin had copied all of Apple's code to ensure perfect compatibility. The Third Circuit decision articulated some fundamental principles on copyright in software programmes. In addition to source code being

[7] *See also*, *Naruto* v. *David John Slater*, (2018); *Chapman Kel*ley v. *Chicago Park District*, 635F.3d 290 (7th Cir. 2011). In this Case, Chapman Kelley is a naturally recognized artist known for his representational paintings of landscapes and flowers –in particular, romantic, floral and woodland interpretation set with ellipses. It is promoted as living art and can the gardener who planted the seeds and saplings maintained it can be considered as the author of the work. "Authors of copyrightable work must be human, work owing their form to the forces of nature cannot be copyrighted"

[8] *See also*, *Feist Publications Inc.* v. *Rural Telephone Service Co.*, 499 US 340 (1991), the case of compilation of facts in a telephone directory can be considered to be copyrightable subject matter. The court interpreted the term originality as "original as the term used in copyright, means only that the work was independently created by the author (as opposed to copied from other works), and that it possesses at least some minimal degree of creativity."

[9] *See also*, *Cartoon Network LP*. v. *CSC Holdings Inc.*, 536F.3d 121 (2d. Cir. 2008); *Williams Electronics Inc.* v. *Artic International Inc*., 685 F.2d 870 (1982). Plaintiff manufactures and sells electronic video games which consist of a electronic circuit board including a microprocessor and memory devices called ROM containing thousands of data and which store instruction and data of a computer programme. In the video game of Defender there are symbols of spaceship and aliens who battle with the symbols of human figures. Obtained copyright for the computer programme and audio-visual effect of the attract mode and play mode. The defendant alleged that the symbols do not comply with the standard requirement of fixation. The symbols of same shapes, size and colours come and go and it is transitory and controlled by the player fails to comply with the requirement of fixation. The court held that "the fixation requirement is met whenever the work is "sufficiently permanent or stable to permit it to be …reproduced, or otherwise communicated" for more than a transitory duration." The audio-visual work is permanently embodied in a material object, the memory devices, from which it can be perceived with the aid of the other components of the game.

[10] Title 17 of the United States Code, sec. 102 (a).

[11] Title 17 of the United States Code, Sec 101 defines Computer Program.

[12] TRIPS Agreement, Art 10.1 - Computer programs, whether in source or object code, shall be protected as literary works under the Berne Convention (1971).

[13] *Apple Computer Inc.*, v. *Franklin Computer Corp.*,714 F.2d1240 (3rd Circuit 1983).

copyrightable as a literary work, the Third Circuit ruled that the object code is also copyrightable as a literary work. The court reasoned that 'the category of literary works…' is not confined to literature in the nature of Hemingway's "*For Whom the Bell Tolls*". The definition of literary work in Section 101 of Title 17 of the United States Code *includes expression not only in words but also 'numbers, or other numerical symbols or indicia',*[14] thereby expanding the common usage of literary works. The Court also rejected *Franklin's* argument that operating system programs cannot be copyrighted, even if applications programs can be. *Franklin* had argued that operating system programs were unprotectable as a system or process pursuant to Sec 102(b) of the Title 17 of the United States Code. The court thought there was no material distinction between operating system programs and application in this regard, in that both instruct the computer to do something. This argument was settled finally in the TRIPS Agreement under Article 10.1, *Computer programme, whether in source or object code, shall be protected as literary work.*

Non-literal or functional element if it contains an expression demarcated from an idea that can be protected under copyright law provided the expression is an original, independently created and not copied from others.[15] While extending protection the doctrine of merger[16], idea-expression dichotomy and doctrine of useful article[17] has to be considered to prevent protection of functional aspects.

[14] Title 17 of the United States Code, Sec 101 defines "Literary works" are works, other than audio-visual works, expressed in words, numbers, or other verbal or numerical symbols or indicia, regardless of the nature of the material objects, such as books, periodicals, manuscripts, phono-records, film, tapes, disks, or cards, in which they are embodied.

[15] Feist Publications, Inc., v. Rural Telephone Service Co 499 US 340 (1991), Interprets Originality as *"the sine qua non of copyright is originality. To qualify for copyright protection, a work must be original to the author. Original as the term is used in copyright, means only that the work was independently created by the author as opposed to copied from other works, and that it possesses at least some minimal degree of creativity. To be sure, the requisite level of creativity is extremely low, even a slight amount will suffice."*

[16] *Baker* v. *Selden*, 101 US 99 (1879) laid down the Doctrine of Merger. Copyright of the book, entitled "Selden's Condensed Ledger or Book-Keeping Simplified" extending copyright to the book will not be extended to the art which is involved in it. Selden's ledger sheets, therefore, enjoyed no copyright protection because they were "necessary incidents to" the system of accounting that he described.

[17] *Mazer* v. *Stein*, 347 US 201 (1954) - copyrightable expression in a useful article, "copyright law does not protect the pictorial, graphic or sculptural features of a useful article, unless those features are 'separable' from the useful article's utilitarian aspects".

Computer programme is considered as an useful article[18] as it performs the function. The functional element being a useful article is protected under the Patent Law[19] but if there is an expression which can be separated it is protected under the Copyright law. Extending protection to the expression part of the programme ensures that too much protection will affect the network effect of software and impede progress in the software industry.

When claim arises for copyright infringement based on copying non-literal elements of computer programme, court had to demarcate the idea and process from expression in a computer programme to the know the exact scope of copyright protection. The structure-sequence-organisation of the non-literal element is considered as the expression that is protected under the copyright law. In writing a programme the programmer has identified the purpose and the ultimate function of the programme, and has also divided the purpose into sub-task with each task being a programme of its own doing a separate function. This is known as a module or routine. Each sub-task is further divided into sub-modules or sub-routines and the efficiency of the programme lies in the way the arrangement of each module and sub-modules are done. These arrangements can be done in various ways which is considered as the total look and feel of the computer programme protected under the copyright law.

The Court has taken into account the software's unique technical and economic features in establishing the line between protectable expression and unprotectable ideas and processes. In *Computer Associates International* v. *Altai Inc.*,[20] the court has laid

[18] Title 17 of the United States Code, Sec 101 defines a 'useful article' is an article having an intrinsic utilitarian function that is not merely to portray the appearance of the article or to convey information. An article that is normally a part of a useful article is considered a 'useful article'.

[19] *See also*, *W. C. M Baker* v. *Charles Selden*, 101 U S 99 (1879) states that "*The description of the art in a book, though entitled to the benefit of copyright, lays no foundation for exclusive claim to the art itself. The object of one is explanation, the object of the other is use. The former may be secured by copyright. The latter can only be secured, if it can be secured at all, by letters patent…*".

[20] *Computer Associates International* v. *Altai, Inc.*, 982 F.2d 693 (2d Cir. 1992), CA's marketed programme CA-Scheduler is a job scheduling programme designed for IBM mainframe computers. When the demand for the programme increased in other operating system instead of rewriting the programme compatible to the other system. They developed a sub-program entitled Adapter, is an "operating system compatibility component" which means it serves as a translator. Altai developed its own independent programme Zeke, which was designed for use with a VSE operating system. In response to customers' demand Altai decided to rewrite Zeke so that it can be run in conjunction with an MVS Operating system. The task was assigned to Arney, who worked in CA and involved in the writing of Adapter. He made a programme similar to Adapter by copying the source code and created Oscar 3.4. After knowing that Arney was involved in the development of Adapter, Altai reconstituted the team of programmers to write a programme similar to Oscar 3.4 and created Oscar 3.5. CA files a case against Altai stating the infringement of their Adapter programme by the Oscar 3.5, the contention

down the test of Abstraction-Filtration-Comparison to demarcate the copyrightable expression from non-copyrightable expression.

The Court would first break down the allegedly infringed programme into its constituent structural part. Then, by examining each of these parts for such things as incorporated idea, expression that is necessarily incidental to those ideas, and elements that are taken from public domain, the court would filter out all the non-protectable material left with a kernel called the 'Golden Nugget' comprising of protectable expression and compare the same with the allegedly infringed programme. Applying the test, the court is of the opinion that nothing is there in the *Oscar* 3.5 which is a protectable expression similar to the *CA-Adapter*. Computer programmes are considered as useful articles which perform functions and have to be protected under the Patent or Trade Secret rather under the Copyright Law, but still protected as a literary work conferring incentive to the creator/developer of the programme and to stimulate the competitive industry.

2.1. Menu Command Structure/Hierarchy

The third condition which the statute set forth limitations on the works that can be copyrighted like copyright protection cannot be extended to *"any idea, procedure, process, system, method of operation, concept, principle or discovery*".[21] Application Programme Interface (API) can be protected under copyright law, API means an interface by which the user interacts with the hardware and the software. In the case of *Lotus Development Corp.* v. *Borland International Inc.*,[22] the court has to decide whether a computer menu command hierarchy is copyrightable subject-matter. *Borland* copied the menu command of Lotus 123 spreadsheet programme contending that the menu commands are the method of operation and excluded from copyright protection under section 102(b) of the 17 USC. The First Circuit held that Lotus menu command hierarchy is a non-copyrightable 'method of operation' and the 'expressive' choices of what to name the command terms and how to arrange them do not magically change the non-copyrightable menu command hierarchy into copyrightable subject-matter.

of Altai that there is no literal copying of the source code of Adapter, Oscar was created independently and not copied. But CA is of the opinion that the Altai has copied the non-literal element of structure sequence organization. The Court has to separate the expressive part from the non-literal element to see whether there is copying of copying expression.

[21] Title 17 of the United States Code, Sec. 102(b).

[22] *Lotus Development Corp.* v. *Borland International Inc.*49 F.3d807(1st Cir. 1995).

In *Oracle America, Inc.*, v. *Google Inc.*,[23] the Federal Circuit had taken a stand different from the First Circuit while discussing whether *Google* by copying the 37 package of the declaring code of Java API for programmer compatibility is an infringement. *Google* copied the declaring code and created its own implementing code, which consists of the step-by-step instructions the computer follows to carry out the declared operation. *Oracle's* declaring code names each operation (or methods) contained in the APIs and defines how the methods are organised within Java.

The Federal Circuit held that the Java declaring code was indeed a method of operation but held that "the copyright protection accorded a particular expression of an idea is not extinguished merely because that expression is embodied in a method of operation". In Federal Circuit's view, if the implementation of a particular method of operation involves some creative choice, then it will be copyrightable. The code can be expressed in different ways, as such *Oracle* created the API packages of 'java.lang.Math.max', organising the similar task or method in a same class, and similar class in the same package like the file cabinet-drawer-file in identifying the particular task to be implemented using the implementing code. Nothing prevented *Google* from writing its own declaring code, along with its own implementing code similar to how *Apple* and *Microsoft* created their own declaring codes for smartphones.

Comparing the two cases, in *Lotus's* First Circuit is of the view that the menu command structure are method of operation and excluded from copyright protection but in *Oracle's* case the Federal Circuit is of the view that only because menu commands are method of operation not excluded from copyright protection, if it contains an expressive element that is copyrightable. *Borland* copied the menu command of Lotus 123 to make them compatible so that the users who are using *Lotus* can easily use *Borland Quattro* without learning new commands and macros. *Google* copied the Java API declaring code in the Android mobile operating system to achieve some level of compatibility for the programmer. The purpose is for interoperability in both the cases but interpreted differently by the court. On appeal the US Supreme Court discussed the fair use concept in *Google LLC*. v. *Oracle America, Inc.*,[24] approving the *Lotus* stating the declaring code

[23] *Oracle America, Inc.*, v *Google Inc*.750 F.3d1339 (Fed. Cir. 2014)
[24] *Google LLC* v. *Oracle America, Inc.*141 S. Ct. 1183 (2021).

is method of operation excluded from copyright protection under section 102 (b) of 17 USC and the interpretation posed by the Federal Circuit is of no use.

2.2. Fair Use of Copyright Work

Another limitation on the copyright work is fair use, the doctrine originating in the courts as an equitable rule of reason that permits courts to avoid rigid application of the copyright statute when on occasion, it would stifle the very creativity which that is designed to foster.[25] The statutory provision that embodies the doctrine indicates, rather than dictates, how courts should apply it. The provision says:

> *"the fair use of a copyrighted work,…for purposes such as criticism, comment, news reporting, teaching…scholarship, or research, is not an infringement of copyright. In determining whether the use made of a work in any particular case is a fair use the factors to be considered shall include-*
>
> *(1) The purpose and character of the use, including whether such use is of a commercial nature or is for non-profit educational purposes;*
>
> *(2) The nature of the copyrighted work;*
>
> *(3) The amount and substantiality of the person used in relation to the copyrighted work as a whole;*
>
> (4) *The effect of the use upon the potential market for or value of the copyrighted work".*[26]

The factors for determining the fair use are not exhaustive and each factor has to be weighed to determine whether the work is a fair use or infringement of copyrighted work. The Fair Use doctrine is flexible to accommodate the changes in technology by upholding the balancing nature of copyright statute. The court in the *Campbell* case laid down the test of transformative to consider whether the new work is a fair use of the original work. If the copied work is commercial in nature it is considered as the unfair use of the copyrighted work but if it is resulting in transformative considered to be fair use. Creation of transformative work further the objective of copyright law to promote science and arts. Transformative means *"whether the new work merely supersede the objects of original creations or instead add something new, with a further purpose or different character, altering the first with new expression, meaning or message; it asks,*

[25] Stewart v. Abend 495 US 207, 236 (1990).

[26] Title 17 of the United States Code, Sec107.

in other words, whether to what extent the new work is transformative." Prior to *Campbell* case, in *Sony*[27] dealt with the test of productive use wherein the borrowed work if it is productive in nature then the exception of fair use will apply.

In *Luther R.Campbell* v. *Acuff-Rose Music, Inc*.,[28] Member of the rap group 2 Live Crew composed and released the rap parody song, 'pretty women', based on the Roy Orbinson song, "Oh Pretty Woman", after Acuff-Rose Music Refused to grant the band a license to produce parody. A year after the song was released and over 2,50,000 copies sold, Acuff-Rose Music sued 2 Live Crew and its record company, Luke Skywalker Records, for copyright infringement. The Court had to look into the followings issues –

i. *Does the commercial nature of the parody provide reasonable unfair use of copyrighted material?*

ii. *Did 2 Live Crew take too much from the original song to warrant fair use (third factor)?*

iii. *Was there market harm due to commercial uses (fourth factor)?*

The US Supreme Court decided that 2 Live Crew's Parody was within the fair use. *Souter J*, believed that the amount of content copied was reasonable in relation to creating a parody of "Oh Pretty Woman". But copying the original line and riff from the song (heart of the song) was necessary when it came to making the parody and was sufficiently different in subsequent lyrics to create a new transformative work. The commercial nature of the parody is just one element to be considered. Each case must be reviewed on a case-by-case basis on the four factors of the Copyright Act. The factors should not be assessed in isolation but all the factors are to be explored, and the results are weighed together, in light of the purposes of the copyright. The goal of copyright, to promote science and the arts, is generally furthered by the creation of transformative works. Such works thus lie at the art of the fair use doctrine's guarantee of breathing space within the confines of copyright, and the more transformative the new work, the

[27] *Sony Corporation of America* v. *Universal City Studios, Inc*., 464 US 417 (1984), in this case Sony manufactured Betamax DVD player and it is used for recording the programme when the viewer is not in position to watch when telecasted and can watch later in their convenient time. This is called time shifting and considered as a fair use by applying the test of transformative.

[28] *Luther R.Campbell* v. *Acuff-Rose Music, Inc*., 510 US 569(1994).

less will be the significance of other factors, like commercialisation, that may weigh against a finding of fair use**.**

The Court in subsequent cases adopted the transformative test laid down in *Campbell*, to substantiate the new work is transformative, by applying the transformative as new insights, new creative metamorphosis and a new purpose. To be considered as a parody it's not based on the self-proclaimed parodist view, but how the consumer receives the defendant's use or based on the expert view.

In the case of *Leibovitz* v. *Paramount Pictures Corp.*[29], Leibovitz had photographed a pregnant and nude Demi Moore for the cover of the August 1991 issue of Vanity Fair Magazine. The photograph attracted a significant amount of public attention, and that issue became a top seller for Vanity Fair. Paramount Pictures, distributor of the film '*Naked Gun33-1/3: The Final Insult*' released a promotional ad poster for its upcoming release, with star Leslie Neilson's face superimposed on the body of a nude, pregnant model posed in the same position as Moore and the tagline "Due this March". Paramount poster to be a parody and considered as a fair use, it reasoned that "*the ad adds something new and qualifies as a 'transformative' work. The smirking face of Nielsen contrast so strikingly with the serious expression on the face of Moore, the ad may reasonably be perceived as commenting on the seriousness, even the pretentiousness, of the original.*" The contrast achieves the test of ridicule and serves as a sufficient comment to be considered as fair use.

In the case of *Suntrust Bank* v. *Houghton Mifflin Co*,[30] the copyright owner of '*Gone with the Wind*', one of the World's best-selling books, the fictional story of Scarlett O'Hara, the spoiled daughter of a wealthy Southern plantation owner who tries to escape poverty after the American Civil War. Alice Randall wrote a book titled '*The Wind Done Gone*', a fictional work based on '*Gone with the Wind*'. In the book, Randall appropriates the character, plots and major scenes from '*Gone with the Wind*' to tell the alternative account of '*Gone with the Wind*'s story from the point of view of one of O'Hara's slaves, Cynara, and the daughter of O'Hara's father and Mammy, a slave who was O'Hara's childhood nurse. She claims her work is not a general commentary upon the Civil-War-era American South, but a specific criticism of and rejoinder to the depiction of slavery

[29] *Leibovitz* v. *Paramount Pictures Corp*., 137F.3d 109, 114-115 (2nd Circuit 1996).
[30] *Suntrust Bank* v. *Houghton Mifflin Co.*, 268 F.3d 1257, 1259 (11th Circuit 2001).

and the relationship between blacks and whites in '*Gone with the Wind*'. Randall chooses to convey her criticism of the work and gives a more powerful message in her work and this work is considered as a fair use under the test of transformative.

In both the cases it gives a critique of the original work and a different or new message and new purpose, for doing this they copied the heart of the original work. Their substantial copying is allowed under the test of transformative as an insight of the original and considered as a fair use.

In a subsequent case of *Cariou* v. *Prince*,[31] relating to appropriation art, Richard Prince appropriated the photograph of Plaintiff Patrick Cariou which was clicked by him while he spent six years living with Rastafarians in Jamaica. It was published in a book '*Yes Rasta*' where Cariou's serene and deliberately composed portraits and landscape photographs depict the natural beauty of Rastafarians and their surrounding environment. Prince has created a collage on canvas that incorporates colour, features distorted human and other forms. Cariou emphasised the aesthetic form of the photograph but Prince appropriated the art not with the intention to have the same approach as like a regular photograph, but a disfigured human photo and depiction of the work in crude and jarring form. Prince made a kind of fantastic, absolutely hip, up to date, contemporary take on the music scene. The Second Circuit, observed that Prince's artwork is of transformative nature, has drastically different approach and is aesthetic from the original work. Hence the Court held Prince's photograph as a fair use of creative metamorphosis which makes it transformative from the original work.

In yet another case, the Court is liberal in interpreting the provision of fair use and considering the work is transformative if the purpose is different from the original copyrighted work. As in *Kelly* v. *Arriba*[32] and *Perfect 10* v. *Amazon*,[33] using the software crawl copied the full size of the image available in the website to create a thumbnail image for the purpose of providing access and sharing the link through their image search engine. Any search for a particular page using the image search engine, provided the thumbnail with the html link and the browser can access the full size image by accessing the website where the full size image is uploaded. The thumbnail image is a small size

[31] *Cariou* v Prince, 714 F.3d 694, 706-07 (2d Cir. 2013).

[32] *Kelly v Arriba Soft Corp*., 336 F.3d 811 (9th Cir. 2003).

[33] *Perfect 10, Inc. v Amazon.Com.*, 508 F.3d 1146 (9th Cir. 2007).

image with lower resolution, if any one wants to enjoy the aesthetic feature of the image they have to see the full size image. The Court held that the thumbnail image is transformative and has a purpose different from the original copyright work which cannot be a substitute for the original work. The court permitted even the substantial/whole copy of the full size image.

In the *Authors Guild* v. *Google, Inc.*,[34] Google in its Google Book maintained the index of more than 20 million books. To create an index, they scanned the entire book which included both the copyrighted work and the works in the public domain and made a digital copy of the same to enable the search so as to create a snippet for the same. The snippet is nothing but the three lines of the page of the book which will be provided in accordance with the search for particular words. If snippets serve the purpose of request and prevent purchasers from purchasing the book, then such snippets will be black listed. The Court observed that the defendant's creation of a searchable 'digital corpus' comprising scanned copies of tens of millions of books that enabled researchers, scholars and others to pinpoint the exact page of any book in the catalogue on which the searched term was used was a 'quintessentially transformative use' and considered as a fair use.

The jurisprudence of fair use doctrine allows the substantial copying if the resulting work is transformative with a new character or message and not a substitute for the original work. Transformation is also included in the derivative work and it is protected as the right of the copyright owner. Then creating a transformative work by substantial copying is not the derivative work protected by the statute. The responsibility of the court while allowing the transformative work as a fair use has to ensure that the new work is not the derivative of the earlier work. The line separating the transformative work as a fair use and derivative work as a right of the copyright owner has to be properly identified. By expanding the jurisprudence of fair use the Court is encroaching on the derivative work and depriving the incentive to the copyright owners.

3. Software Interoperability

The Court, while assessing the fair use of the original work, considered whether the work is creative or functional. If the work is more informative or functional than creative it favours fair use. Computer programme is predominantly functional, courts

[34] *Authors Guild* v *Google, Inc*., 804 F.3d 202 (2d Cir. 2015).

allow the fair use as stated in the Campbell case if the copyrighted work is far "from the core of copyright" and it is "inherently bound with non-copyrightable ideas". The value of the code depends on how it works with other codes, each programme is compatible and interoperable to each other and for this the developer of the code should know the interface/ functional element.

Interoperability is defined as "the logical and where appropriate, physical interconnection…to permit all elements of software and hardware to work with other software and hardware and with users".[35] There are two Ninth Circuit decisions involving reverse engineering software for interoperability as fair use if the resulting use results in intermediate copying.

In the first Ninth Circuit case, *Sega Enterprises Ltd.* v. *Accolade, Inc.,*[36] *Accolade* decompiled the Genesis Console of *Sega* to know the underlying functional element so that the independently created video game cartridges can run with the console of *Sega*. Computer programmes swathe both the copyrightable and non-copyrightable elements and to discern the non-copyrightable functional elements decompilation is the only means that results in the intermediate copying. If decompilation is not allowed it will result in *de facto* monopoly over the functional elements under the copyright law. For the purpose of interoperability, decompilation is allowed as a fair use provided the necessary element is not readily available in any other form.

In the second Ninth Circuit case, *Sony Computer Entertainment Inc.* v. *Connectix Corp.,*[37] Connectix decompiled the Sony PlayStation BIOS to know the underlying functional element inorder to create the Virtual Game Station that emulates the same as the Sony PlayStation. It is possible to play video games without the Sony PlayStation Console and TV sets, if the computer downloads the software containing the CD –Drive. The Court allowed the decompilation of the BIOS/Firmware results in intermediate copying, as a fair use for the purpose of interoperability and the resulting work is modestly transformative though the use is commercial.

[35] EU Software Directive, Recital 10, *available at*: https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX%3A32009L0024 (last visited on September 15, 2022).

[36] *Sega Enterprises Ltd.* v. *Accolade, Inc.*, 977 F.2d1510 (9th Cir. 1993).

[37] *Sony Computer Entertainment Inc.* v. *Connectix Corp.*, 203 F.3d 596 (9th Cir. 2000).

3.1. Google LLC v. Oracle America, Inc.[38]

Oracle, the owner of Java SE was created by Sun Company as a software platform using the Java programming language. Applying the programming language, the users (programmers) create application programmes for the desktop and laptop. *Google* acquired Android, the start-up firm in smartphone software as they needed a new platform to be used in mobile devices like smartphones. They copied the Java SE API for achieving interoperability so that the users (programmers) who spent considerable time in learning the Java language will find it easier to develop application programmes for the new platform.

The API of the *Oracle* Java SE includes the Implementing Code, a step by step instruction, which enables a computer programme to perform the task by executing the code. For each task there is a separate implementing code which varies from simple task to complex task. Google did not copy the implementing code but they created their own.

To identify which implementing code has to be adopted to execute a particular task the developers of programmers developed specific commands referred to as 'Method calls'. The example given by the District Court to explain the precise technology of method call – 'java.lang.math.max'. As part of his/her software, a programmer wishes to discover which of two integers is greater. To do so in Java, the programmer must first write the words java.lang. Those words (in bold) allude to the 'package' (or, as a metaphor analogy, the file cabinet). The programmer will then write Math which refers to the 'class' (or, to use an analogy, the drawer). He or she will then write max which refers to the 'method' (or, to use an analogy, the recipe). After that, the programmer will add two brackets () and between the brackets, the programmer will enter two integers, say 4 and 6, to compare. This is how the entire expression, or method call, will appear: 'java.lang.Math.max (4, 6)'. The use of this expression will invoke a task-implementation programme that will choose the greater number using the API.

Next is the Declaring Code, it is a link between the 'Method Call' and the 'Implementing Code'. Declaring Code labels the particular tasks in the API and organises those tasks, or methods into packages and classes. 37 packages of the declaring code have been verbatim copied by *Google* in the new platform for mobile device of smartphone for

[38] *Google LLC* v. *Oracle America, Inc.*141 S. Ct.1183 (2021).

the purpose of user interoperability. Oracle does claim that Google's use of the Sun Java API's declaring code violates its copyrights.

Google did not copy the implementing code but only the declaring code consisting of 11,500 lines of codes which goes to 600 pages. Instead of copying the declaring code, *Google* should have written its own declaring code similar to implementing code just like Apple and Microsoft companies.

The Court had to decide whether copyright law could protect Java's API and whether *Google's* use of *Oracle's* API infringed its copyright and if so, whether the fair use defence can be applied.

The Court assumed that the entire Sun Java API falls within the definition of that which can be copyrighted. This was criticised by the minority that before proceeding with the fair use argument, the majority should look into the copyright protection of Java's API. Previously in *Lotus* v. *Borland* case, the Court observed that the menu command is considered as the method of operation and excluded from copyrighting under sec. 102 (b) of 17 United States Code. *Google* contended that the API's declaring code and organisation fall into the category of method of operation and expressly excluded from copyright protection. Federal Circuit in *Oracle* v. *Google*, held that the expressive element in the code can be protected under the copyright statute. *Oracle* in the declaring code labels the task in the API and organises the same as a method-classes-packages.

3.2. The four factors of Fair Use

The question arises whether Google's copying of the Sun Java API platform in the mobile device of smartphone, specifically verbatim copying the declaring code and the organisational structure for 37 packages of that API, a fair use. The Court had to consider the four factors set forth in the section 107 of the statute whether copying is allowed as a fair use.

3.2.1. The Nature of the Copyrighted Work

Federal Circuit	SC-Majority	SC- Minority
The declaring code and the SSO of the 37 API Packages at issue were	*Google* used creativity to develop Android for use in smartphones. It copied	Declaring code is closer to the core of copyright Developers cannot even

sufficiently creative and original to qualify for copyright protection, but functional considerations were substantially and important - Favours fair use	the declaring code which is considered to be far from the core of copyright than the implementing code and inherently bound together with non-copyrightable ideas. If the work is more informational and functional - Favours fair use.	see implementing code …Implementing code thus conveys no expression to developers. Declaring code, in contrast, is user facing - Neutral or even fair use

3.2.2. The Purpose and Character of the Use

Federal Circuit	**SC-Majority**	**SC- Minority**
Use was commercial. *Google* stands to profit from exploitation of the material without paying the customary price - Against fair use	*Google* copied them because programmers had already learned to work with the Sun Java API's system, and it would have been difficult, perhaps prohibitively so, to attract programmers to build its Android smartphone system without them. So he copied the declaring code verbatim of 37 packages both the labelling of the methods and the organizing of the methods as classes and	2015 alone, *Google* earned $18 billion and *Google* used code for the exact same purpose as *Oracle*. *Google* copied the declaring code verbatim the naming the organisational structure for the same purpose as *Oracle* and cannot be considered as transformative but derivative work which is an infringement of the rights of copyright owner under section 106 (2) - Against Fair Use

	packages. The copying is considered as transformative using the code in a new platform of smart-phones – Favours fair use	

3.2.3. The Amount and Substantiality of the Portion Used

Federal Circuit	SC-Majority	SC- Minority
The parties stipulated that only 170 lines of codes were necessary to write in the Java language. It is undisputed, however, that *Google* copied 11,500 lines of codes - 11,330 more lines than necessary to write in Java. That Google copied more than necessary weighs against fair use - Neutral or even against fair use	The better way to look at the numbers is to take into account the several million lines that *Google* did not copy. Did not agree that *Google* could have achieved its Java Compatibility by copying only the 170 lines of codes - Favours fair use	*Google* does not deny that it copied the heart or focal points; "the declaring code is what attracted programmers to the Java platform"; "A copied work is quantitatively substantial it could serve as a market substitute for the original work or potentially licensed derivatives of that work" as stated by Justice Campbell - Against fair use

3.2.4. Market effects

Federal Circuit	SC-Majority	SC- Minority
"The record contained substantial evidence that Android was used	"*Google's* Android platform was part of a distinct market than Java	*Google* eliminated manufacturers 'willingness to pay for

as a substitute for Java SE and had a direct market impact…" that *Oracle* never built a smartphone device is irrelevant because potential markets include licensing others to develop derivative works. - Against fair use	Software…Android platform…offers 'an entire mobile operating stack', is a very different type of product than Java SE, which is just an applications programming framework". – Favours fair use	Java; after Android release, Amazon used the cost free availability of Android to negotiate it 4X; *Microsoft*/*Apple* developed their own mobile system'. Potential market effect means not only the creator of original works would in general develop but also the copyright holder might license others to develop. – Against fair use.

4. Conclusion

Google copied the declaring code of API verbatim comprising 37 packages consisting of 11,500 lines of codes about 600 pages but still the majority in the case considered it only 0.04% of the entire API. Majority is of the opinion that the declaring code is far from the core of copyright protection than the implementing code which is closer. For a Programmer, while writing the application programme for a new platform, it is the declaring code which matters and not the implementing code which is not visible. Every copyrightable work is inherently bound with non-copyrightable ideas, what is protected is only the expression the same is the concern with respect to the code of the programme. Substantial copying of the Java SE API and using it as the same but in a new device will not result in the creation of new work or transformative but a derivative work. Court while interpreting the factors of the fair use, should consider whether the work is transformative or derivative work where both are transformation or adaptation of the original work. While considering the factors of fair use, each factor should be given proper weightage and not by stating if the work is transformative though commercial use is still fair under the first factor. Once considered the work is transformative the other factor not having proper weightage is not justifiable. Each factor is considered to have its own relevance while considering whether the substantial copying is transformative or derivative of the original work.