

High Court Oracle Copyright Ruling Is A Boon For Innovation

By **Dorothy Auth, Howard Wizenfeld and Jaclyn Hellreich**

(April 8, 2021, 4:48 PM EDT)

On April 5, the U.S. Supreme Court ruled 6-2 in favor of Google in the case of *Google LLC v. Oracle America Inc.*,^[1] a landmark case for computer software protections under U.S. copyright law. The court addressed the applicability of the copyright law to protecting a software interface, also known as an application programming interface, or API, particularly in the context of whether Google's use of another's API to create new computer programs constitutes fair use.^[2] This decision creates law useful in guiding software engineers on how to develop new computer code using existing APIs, thereby supporting further growth in the high technology space.

The program at issue is called Java, which is a popular computer language with widespread utilization in commercial programs. Google wanted to use the Java programming language in developing its Android operating system for use in mobile phones. To function properly, the Java language requires the use of the Java API, a library of prewritten functions organized into packages that include declaring code and implementing code. Declaring code provides the name of the task, e.g., print, requested by a programmer and the location for the implementing code, whereas implementing code tells the computer how to execute the requested task, e.g., code providing step-by-step instructions to carry out a print request.^[3]

In its efforts to develop its Android operating system for mobile phones, Google copied the declaring code of Java APIs into Android on the theory that the declaring code was not copyrightable.^[4] Google wrote its own implementing code for use in Android's mobile phone operating system, which took approximately 100 Google engineers over three years to develop.^[5] In 2010, Oracle sued Google for copyright infringement due to use of Java APIs in Android devices.^[6] The U.S. Court of Appeals for the Federal Circuit found that Google had infringed Oracle's copyright in Java and had no defense of fair use. Google appealed to the Supreme Court.

Before this case, the question of copyright protection for an API or software interface was in doubt. Besides the Federal Circuit decision in the *Google* case, only four circuit courts^[7] had previously



Dorothy Auth



Howard Wizenfeld



Jaclyn Hellreich

discussed copyright protection for computer software, and those cases were arguably distinguishable from, or even irrelevant to, the scenario presented by the *Google* case. The U.S. Court of Appeals for the First Circuit's 1995 *Lotus Development Corp. v. Borland International Inc.* decision appears as one of the only circuit court cases similar in fact to that of *Google*.

The First Circuit in *Lotus* held the function and arrangement of commands in so-called menu trees found in the Lotus 1-2-3 spreadsheet program were not copyrightable.[8] Users of Lotus 1-2-3 could create macros by using menu trees, providing a command hierarchy, containing commands such as copy, print and quit, which in turn called a further code to carry out the identified function.[9] This menu command hierarchy was held to be an uncopyrightable method of operation because the hierarchy provided "the means by which users control and operate Lotus 1-2-3 ... [and] serve[d] as the method by which the program is operated and controlled." [10] The *Lotus* court distinguished the hierarchy from underlying computer code because "while [the underlying computer] code is necessary for the program to work, its precise formulation is not." [11] Indeed, the menu command hierarchy was held to be functionally required "to allow users to operate its programs in substantially the same way." [12] The copyright holder argued the menu hierarchy was copyrightable because the choice and arrangement of its command terms were expressive. [13] However, the First Circuit found this argument irrelevant because these choices and arrangement were still part of Lotus 1-2-3's method of operation, holding this code uncopyrightable. [14]

In contrast, the Federal Circuit — a court that does not traditionally hear copyright cases [15] — held the functional declaring code in Java's API was copyright protectable in the *Google* case. [16] The Federal Circuit stressed that Oracle had "unlimited options as to the selection and arrangement" of the API and, therefore, that the function and arrangement of the declaring code was expressive and copyrightable. [17] *Lotus*' analysis was rejected because the Federal Circuit believed copyright protection should not be denied due to an abstract creative expression being tied to a function. These decisions, although not completely analogous, created a circuit split of sorts, leaving the copyrightability of such API code in question.

Other than *Lotus*, there were arguably no other circuit decisions quite on point for Google's petition for certiorari to the Supreme Court. [18] As the Supreme Court generally waits for a clear circuit split on an issue before granting certiorari, it was unclear whether it would grant Google's petition merely on a potential split between the First and Federal Circuits. However, one could see how the preprogrammed functions discussed in *Lotus* were similar to the functional declaring code at issue in the *Google* case. The menu command hierarchy and declaring code are both user facing and allow users to call on longer strings of code without having to expressly type in each command. As in *Lotus*, where the hierarchy's value was tied to users' familiarity with it, the value of Java is also based on its universal adoption as a popular computer language. [19] With that said, both Oracle and the U.S. solicitor general denied that a circuit split was present and argued that the lack of available circuit precedent on this warranted denial of Supreme Court intervention. [20] Despite these arguments, the Supreme Court granted certiorari.

The Supreme Court in *Google* was mindful of not unduly burdening users of software programs with overbroad copyright protection and chose to assume copyrightability, without making any express ruling on this threshold issue. The court focused its efforts instead on deciding whether copying the declaring code was permitted under the fair use doctrine, [21] and the court recognized that this doctrine is an important tool in determining the lawful scope of computer program copyrights. [22] The doctrine distinguishes between expressive features of computer code that are copyrightable and functional features that are not, with some types of software falling somewhere in between as held in *Google*. [23] In considering the type of copyrighted work, the majority indirectly opined on the copyrightability of the

API, holding that declaring code differs from other types of computer code because it is "inextricably bound" to "the division of computing tasks" and the idea of organizing tasks into groups, neither of which is the traditional subject of a copyright.[24] As a result, the court did not afford declaring code the same level of copyright protection as other computer programs, e.g., implementing code,[25] stating that "declaring code is, if copyrightable at all, further than are most computer programs (such as the implementing code) from the core of copyright." [26]

Justice Clarence Thomas, joined by Justice Samuel Alito, in the dissent ridiculed the bifurcation of copyrightability from fair use and maintained that Congress expressly provided equal protection for all computer code under the Copyright Act because the act expressly defines "computer program" as "'a set of statements or instructions to be used *directly* or *indirectly* in a computer in order to bring about a certain result.'" [27] This statute therefore provides copyright protection over declaring code that *indirectly* provides a computer with instructions for carrying out a task. As a result, the dissent argues the majority distorts the fair use doctrine to apply in a case where "Oracle's code is ... copyrightable, and Google's use of that copyrighted code is anything but fair." [28]

Many companies will look to the Supreme Court's decision in *Google* for guidance on the scope of Java's copyright protection, and some take-home messages are apparent. First, not all software is created equal. Courts following the *Google* decision will likely analyze in detail the fair use doctrine to determine the strength or availability of copyright protection. To the extent that a computer code can be characterized as functional in nature, that characterization will likely narrow its protections under the Copyright Act. In the simple case of declaring code, less copyright protection is conferred than on implementing code, i.e., many uses of declaring code may be considered fair uses. Second, reimplementing of an API may be a fair use of computer code. In other words, where a computer code is substantially rewritten for a different use, e.g., from use in a desktop computer to use in a mobile phone, this may be sufficient to demonstrate fair use of the code. Third, the amount of code copied may impact a determination of fair use and may be measured based on the total size of the overall program copied. *Google* makes clear that this factor considers the entirety of the software when determining the percentage copied, not a percentage of the type of code copied. Finally, whether copied code results in a product sufficiently different from the starting code may influence the fair use determination.

The majority specifically did not overturn *Lotus*, but instead hinted at the incompatibility of the two decisions by saying "applying copyright law to computer programs is like assembling a jigsaw puzzle whose pieces do not quite fit." [29] As a result, many questions remain unanswered. Although Google used the Java API for the development of a mobile phone interface, it remains unclear how or if this case may affect software developers using the same platforms as originally targeted by Java, e.g., desktop computers and laptops.[30] It is also unclear whether the use of original implementing code is essential to the fair use analysis. For example, because Java code is in such wide use, is it permissible to use Java declaring code and implementing code if together their use is for a different purpose? If not, how much of the implementing code must be changed to be considered a fair use? Is there any type of declaring code that could be considered sufficiently nonfunctional or creative to be afforded a larger scope of copyright protection? These questions will require further court decisions to elucidate.

For the time being, the software industry is changing at a lightning-speed pace, and the *Google* decision opens the door for software developers to generate new software based on existing APIs whose underlying implementing code has been rewritten.

Wickersham & Taft LLP.

Howard Wizenfeld is special counsel at Cadwalader.

Jaclyn Hellreich is an associate at the firm.

The opinions expressed are those of the author(s) and do not necessarily reflect the views of the firm, its clients or Portfolio Media Inc., or any of its or their respective affiliates. This article is for general information purposes and is not intended to be and should not be taken as legal advice.

[1] *Google LLC v. Oracle America, Inc.*, No. 18-956, slip op. at 1 (U.S. Apr. 5, 2021) (majority opinion).

[2] *Id.* at 14-15.

[3] *Id.* at 4-5.

[4] *Oracle America, Inc. v. Google Inc.*, 2016 WL 3181206, *7 (N.D. Cal. 2016).

[5] *Google*, slip op. at 3-4 (majority opinion).

[6] *Google*, slip op. at 9 (majority opinion).

[7] The four cases discussed were *Lotus Develop. Corp. v. Borland Intern., Inc.*, 49 F.3d 807, 818 (1st Cir. 1995), *Lexmark Intern, Inc. v. Static Control Components, Inc.*, 387 F.3d 522 (6th Cir. 2004), *Apple Computer, Inc. v. Franklin Computer Corp.*, 714 F.2d 1240 (3d Cir. 1983), and *Computer Ass. Intern. Inc. v. Altai, Inc.*, 982 F.2d 693, 706 (2d Cir. 1992). Other than Lotus, these cases were not discussed in detail by the Court.

[8] *Lotus Develop. Corp. v. Borland Intern., Inc.*, 49 F.3d 807, 818 (1st Cir. 1995).

[9] *Id.* at 809-810 (noting that Lotus 1-2-3 has "69 commands arranged into more than 50 menus...").

[10] *Id.* at 815.

[11] *Id.* at 816.

[12] *Id.*

[13] *Id.*

[14] *Id.*

[15] The case originally involved patent disputes.

[16] *Oracle America, Inc. v. Google Inc.*, 750 F.3d 1339, 1359-1364 (Fed. Cir. 2014).

[17] *Id.* at 1361 (internal quotations omitted).

[18] See n. 7.

[19] Google, slip op. at 2-3 (majority opinion).

[20] See Brief in Opposition at 17, *Google v Google LLC v. Oracle America, Inc.*, (U.S. Apr. 5, 2021) No. 18-956; Brief for the United States as Amicus Curiae at 15, *Google v Google LLC v. Oracle America, Inc.*, (U.S. Apr. 5, 2021) No. 18-956.

[21] Google, slip op. at 2-3 (majority opinion).

[22] *Id.* at 17.

[23] *Id.*

[24] *Id.* at 22.

[25] *Id.* at 24.

[26] *Id.*

[27] Google, slip op. at 4 (quoting 17 U.S.C. §101) (Thomas, J., dissenting).

[28] *Id.* at 2.

[29] Google, slip op. at 16 (majority opinion) (citing *Lotus*, 49 F.3d at 820).

[30] Google, slip op. at 3-4 (majority opinion).