

Content, Media and Entertainment

US Supreme Court Holds that Copying "Declaring Code" Is Fair Use

By: [Steven R. Englund](#), [David R. Singer](#), [Alison I. Stein](#), and [Cayman C. Mitchell](#)

On April 5, 2021, the US Supreme Court decided the long-running and closely-watched case of *Google v. Oracle*, holding that it was a fair use for Google to copy the “declaring code” from the application program interface (API) of Oracle’s Java SE platform when implementing Java for Google’s Android operating system.^[1] The Court sidestepped the question of whether Oracle’s copyright protected such code at all, instead assuming that it did. The Court then conducted a fair use analysis that, depending on one’s perspective, was either an “ordinary application of copyright’s limiting doctrines,”^[2] or “wholly inconsistent with the substantial protection Congress gave to computer code.”^[3] Although the Court went to great lengths to cabin its holding to the very narrow type of code at issue,^[4] the decision is sure to feature prominently in fair use analyses in copyright infringement cases in the years to come.

Background

APIs are used by software developers to simplify the development of complex computer programs by allowing two programs to communicate with each other. Java’s API provides access to a collection of prewritten software components that carry out a large number of specific tasks. For instance, instructing a computer how to manipulate the binary values necessary to multiply two non-integer numbers might take dozens of lines of code. Java’s API allows a programmer to simply type the short word command “multiplyExact” to access an already-written set of instructions to do just that.^[5] The programmer need not ever see the underlying code nor understand how it works. If a programmer had to write such instructions for every single task a program is expected to perform, it “would be difficult, perhaps impossible, for a programmer to create complex software programs.”^[6]

Java’s “declaring code” is the code in the API that identifies the prewritten software components provided by the API and tells the API how to communicate with other programs. By contrast, “implementing code” is the code in the API that actually performs a specific task. To use an example on which the Court and parties relied heavily, to find the greater of two numbers, a programmer in Java might type (or in computer lingo, “call”) the method “Java.lang.Math.max (4, 6).” That’s all a programmer need do; the prewritten code in the API will return “6.” But that line only works because of the declaring code.^[7] The declaring code identifies methods such as “max,” and organizes them inside classes such as “Math,” inside packages such as “Java.lang.” The declaring code also tells that method to expect two integers as inputs (in this case, 4 and 6), as well as the output of the method (in this case, another integer, 6). This is called the “structure, sequence, and organization” of the API.^[8]

No one, including Google, disagreed that the copyright in Java SE protected the *implementing* code—the code that does the work. Further, the Court seemed eager to emphasize that its ruling was not intended to affect such code, noting that “the declaring code is, if copyrightable at all, further than are most computer programs (such as the implementing code) from the core of copyright.”^[9] Here, there was no argument that Google copied Oracle’s implementing code, so a fair use analysis as to the implementing code was “not at issue.”^[10] However, it is also true that no one disagreed that Google could have written its own declaring code had it wanted to; indeed, Apple and Microsoft did that in their own implementations of APIs for their platforms.^[11] Google’s stated reason for not doing so was so that programmers already familiar with the Java programming language could easily write new programs for

Android devices, without having to relearn new commands to do the same tasks.

Google v. Oracle was pending for 10 years and has a complicated procedural history. The trial court initially held that the declaring code was not protected by copyright. On a first appeal, the Court of Appeals for the Federal Circuit reversed, finding that copyright protected the declaring code and remanding for a trial on fair use. Google petitioned for *certiorari*, which the Court denied. On remand, a jury then found fair use, and the trial court held the jury verdict was supported by substantial evidence. On a second appeal, the Federal Circuit found that Google's use of the declaring code was not a fair use. The Supreme Court then agreed to consider both the copyrightability and fair use questions during its 2019-2020 term, but cancelled the originally-scheduled oral argument due to the onset of the coronavirus pandemic. The case was finally argued in October 2020.

The Court's Decision

In an opinion full of analogies both practical (the organization of the declaring code is like a file cabinet) and fanciful (use of the API is like having a robot pick out a recipe), the Court, in a 6-2 decision authored by Justice Breyer, found Google's copying to be a fair use. Justice Thomas authored a dissent, which was joined by Justice Alito.

Had the Court addressed the copyrightability question, it would have been the Court's first pronouncement on the scope of copyright protection for computer programs.^[12] However, the Court declined to do so, assuming "purely for argument's sake" that the declaring code was copyrightable.^[13]

Instead, the Court proceeded directly to the issue of fair use. In doing so, the court first provided an important clarification that as a procedural matter, fair use is a mixed question of law and fact. This means that a jury's findings are respected on matters of fact, such as "whether there was harm to the actual or potential markets for the copyrighted work" or "how much of the copyrighted work was copied," but the ultimate question of fair use is a legal question reviewed *de novo*.^[14]

Section 107 of the Copyright Act provides four factors that courts are to consider in deciding fair use issues.^[15] Unlike most fair use decisions, the Court addressed the factors out of order, prioritizing the second factor: the nature of the copyrighted work. This emphasis is unusual. As the Second Circuit has noted, "[t]he second factor has rarely played a significant role in the determination of a fair use dispute."^[16] As a dissent by Justice Thomas points out, the Court historically has considered the first and fourth factors to be the "most important."^[17] Here, despite assuming the copyrightability of the declaring code, the Court reiterated that the declaring code was "inextricably bound up with the idea of organizing tasks into what we have called cabinets, drawers, and files, an idea that is [] not copyrightable,"^[18] suggesting that its copyright protection was "thin."^[19] "Unlike many other programs," the Court reasoned, the declaring code's "value in significant part derives from the value that those who do not hold copyrights, namely, computer programmers, invest of their own time and effort to learn the API's system."^[20] Because Google copied only the declaring code needed for programmers to use known calls, which "embodies a different kind of creativity" from the implementing code that was not copied, the Court concluded that this factor favored fair use.^[21]

Although the Court proceeded to analyze the other three factors and concluded that they each weighed in favor of fair use, the analysis of each subsequent factor was heavily colored by this initial conclusion:

- As to the first factor—purpose and character of the use—the Court found Google's use to be transformative, a conclusion typically reached when the purpose of the use is different from the original use (such as in a parody of the original work). The Court stated that "since virtually any unauthorized use of a copyrighted computer program" would be for the same purpose, "accomplish[ing] particular tasks," the Court must "go further."^[22] In doing so, it found that the factor favored fair use even though the purpose was admittedly the same, because "Google's use of the Sun Java API seeks to create new products" and "expand the use and usefulness of

Android-based smartphones.”^[23] According to Justice Thomas, this conclusion “transforms the definition of ‘transformative.’”^[24] The Court gave little weight to the commercial nature of Google’s use, something that can cut against a finding of transformative use.^[25]

- The Court found that the third factor—amount of copyrighted work used—favored fair use, because while Google copied approximately 11,500 lines of code, that constituted only 0.4% of the Java SE code, and it was copied for functional reasons.^[26]
- As to the fourth factor—the effect of the use on the potential market—the Court concluded that “the jury could have found that Android did not harm the actual or potential markets for Java SE.”^[27] In essence, the Court believed that because Oracle’s platform was not successful in the smartphone market at the time of Google’s copying, and Oracle’s software may not have been suitable for smartphones, it was reasonable for the jury to conclude that Google did not harm the market for the original work.^[28] In a surprising development for a fourth factor analysis, the court also gave weight to the perceived public benefit that Google’s copying of the declaring code would allow programmers who had invested in learning Oracle’s API to more readily create new programs.^[29]

In the end, the Court held Google’s copying of Oracle’s declaring code to be a fair use as a matter of law.^[30] As a result, Google escaped an infringement judgment that potentially could have resulted in tens of billions of dollars in liability, and Oracle will have no redress for copying that it argued diminished its actual and potential Java licensing revenues. In the process, the Court either justifiably applied fair use “flexibl[y],” weighing different factors differently based on the “context” of thin copyright protection in declaring code,^[31] or, as Justice Thomas put it: improperly “used fair use to eviscerate Congress’ considered policy judgment” concerning copyright protection for computer programs.^[32]

This was the Court’s first fair use decision in over 25 years.^[33] And it highlights the unpredictability of fair use determinations by following the pattern of the Court’s previous three fair use cases, which were all reversed at each stage of appeal.^[34] The Court’s embrace of the concept of transformative use in its last fair use decision sparked a wholesale refiguring of the doctrine in the lower courts.^[35] How lower courts apply *Google v. Oracle* in garden-variety software copyright cases, or more broadly in non-software cases, remains to be seen. For example, in future fair use cases, it will be interesting to see whether the Court’s elevation of the second factor leads to more sensitivity to the nature of the work, whether the Court’s skepticism concerning a litigant’s prospects for competing in a certain market will affect future analysis of market harm, and whether the Court’s analysis of transformative use and public benefit will creep into cases involving less functional works despite the Court’s efforts to limit its holding to declaring code. But no matter what, the case will undoubtedly play a prominent role in argument over fair use for many years to come.



Contact Us



Steven R. Englund

senglund@jenner.com | [Download V-Card](#)



David R. Singer

dsinger@jenner.com | [Download V-Card](#)



Alison I. Stein

astein@jenner.com | [Download V-Card](#)



Cayman C. Mitchell

cmitchell@jenner.com | [Download V-Card](#)

Meet Our Team

Practice Leaders

Andrew H. Bart

Co-chair

abart@jenner.com

[Download V-Card](#)

David R. Singer

Co-chair

dsinger@jenner.com

[Download V-Card](#)

Alison I. Stein

Co-chair

astein@jenner.com

[Download V-Card](#)

[1] *Google LLC v. Oracle America, Inc.*, No. 18-956, slip op. (U.S. Apr. 5, 2021). The majority opinion regularly refers to Oracle as “Sun” because the API was originally created by Sun Microsystems before Oracle purchased the company.

[2] *Id.* at 18.

[3] *Id.* at 1 (Thomas, J., dissenting).

[4] See, e.g., *id.* at 35 (majority op.) (“[W]e have not changed the nature of [fair use] concepts [and] do not overturn or modify our earlier cases involving fair use.”).

[5] See Class Math, Java Platform Standard Ed. 8, Oracle (accessed Apr. 8, 2021), <https://docs.oracle.com/javase/8/docs/api/java/lang/Math.html>.

[6] *Google v. Oracle*, No. 18-956, slip op. at 4.

[7] *Id.* at 5 (“Without that declaring code, the method calls entered by the programmer would not call up the implementing code.”).

[8] *Id.* at 4.

[9] *Id.* at 24.

[10] *Id.* at 4.

[11] See *id.* (“Instead of creating its own declaring code—as Apple and Microsoft chose to do—Google copied verbatim . . . Oracle’s declaring code.”); *id.* at 10 (“Google could have written its own declaring code.”); see also *id.* at 7 (Thomas, J., dissenting) (“Certainly, Apple and Microsoft managed to create their own declaring code.”).

[12] Due to the recusal of a justice, the court was evenly divided in its last foray into that topic in *Lotus Dev. Corp. v. Borland Int’l, Inc.*, 516 U.S. 233 (1996).

[13] *Google v. Oracle*, No. 18-956, slip op. at 15 (majority op.).

[14] *Id.* at 19–20.

[15] The fair use factors are: “(1) the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes; (2) the nature of the copyrighted work; (3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and (4) the effect of the use upon the potential market for or value of the copyrighted work.” 17 U.S.C. § 107.

[16] *Authors Guild v. Google, Inc.*, 804 F. 3d 202, 220 (2d Cir. 2015) (citation omitted).

[17] See, e.g., *Google v. Oracle*, No. 18-956, slip op. at 9 n.5, 11, 15 (Thomas, J., dissenting) (noting that effect on potential market for copyrighted work is the “single most important element of fair use” and that purpose and character of use is “the second-most important because it can prove dispositive” (citations omitted)).

[18] *Id.* at 22 (majority op.).

[19] *Id.* at 15–16.

[20] *Id.* at 24.

[21] *Id.* at 23–24.

[22] *Id.* at 25.

[23] *Id.*

[24] *Id.* at 16 (Thomas, J., dissenting).

[25] See, e.g., 17 U.S.C. § 107 (consideration of the first factor “includ[es] whether such use is of a commercial nature or is for nonprofit educational purposes”).

[26] *Google v. Oracle*, No. 18-956, slip op. at 28 (majority op.).

[27] *Id.* at 31.

[28] *Id.* at 31–35.

[29] See *id.*

[30] See *id.* at 35.

[31] See, e.g., *id.* at 15, 18.

[32] *Id.* at 19 (Thomas, J., dissenting).

[33] See *Campbell v. Acuff-Rose Music, Inc.*, 510 U.S. 569 (1994).

[34] See *id.*; *Harper & Row, Inc. v. Nation Enters.*, 471 U.S. 539 (1985); *Sony Corp. of Am. v. Universal City Studios, Inc.*, 464 U.S. 417 (1984).

[35] See, e.g., *Campbell*, 510 U.S. at 579 (“[Transformative] works [] lie at the heart of the fair use doctrine’s guarantee of breathing space within the confines of copyright, and the more transformative the new work, the less will be the significance of other factors” (citation omitted)).

© 2021 Jenner & Block LLP. **Attorney Advertising.** Jenner & Block is an Illinois Limited Liability Partnership including professional corporations. This publication is not intended to provide legal advice but to provide information on legal matters and firm news of interest to our clients and colleagues. Readers should seek specific legal advice before taking any action with respect to matters mentioned in this publication. The attorney responsible for this publication is Brent E. Kidwell, Jenner & Block LLP, 353 N. Clark Street, Chicago, IL 60654-3456. Prior results do not guarantee a similar outcome.