SULLIVAN & CROMWELL LLP

S&C memo

April 6, 2021

Supreme Court Holds That Google's Copying of Java Software Constitutes "Fair Use"

Reversing the Federal Circuit, Supreme Court Holds That Google's "Limited Copying" of Java SE Was a Transformative Fair Use

SUMMARY

On April 5, 2021, the Supreme Court, in *Google* v. *Oracle America*,[1] held that Google's copying of the Java SE API constituted fair use of copyrighted material, reversing the Federal Circuit. Assuming without deciding that the Java SE API software was copyrightable, the Supreme Court held that: (a) whether the copying at issue was fair use is a question of law for the courts; and (b) Google's copying constituted fair use under the statutory four-factor test because (i) Google copied only "declaring" code and not the "implementing" code for the Java SE API, (ii) Google's use was a transformative reimplementation of Java SE, (iii) Google did not copy a substantial amount of the Java SE API, and (iv) Google's copying created a new mobile operating system market that benefitted the public and that the Java SE copyright owner might not have been able to exploit.

BACKGROUND

Java SE is software platform, incorporating the popular software programming language Java, which enables programmers to more easily write interoperable programs able to run on any desktop or laptop computer. The copyright for Java SE is currently owned by Oracle America, Inc. Part of Java SE is a tool called an Application Programming Interface, or API, that "allow[s] programmers to use . . . prewritten code to build certain functions into their own programs, rather than write their own code to perform those functions from scratch."[2] Essentially, an API includes two types of software, called "declaring" code and "implementing" code. Declaring code permits programmers to use relatively simple commands to direct

the performance of specific tasks; the tasks are then executed by the implementing code. By learning just the Java API commands embodied in the declaring code, programmers can relatively easily and flexibly utilize the vast library of implementing code in Java SE to develop new programs.

In 2005, Google acquired Android, Inc., and began to develop its own software platform for mobile devices. Roughly 100 Google engineers spent more than three years writing millions of lines of code, optimized to smartphone technology. Although Google wrote entirely new implementing code, it also copied 11,500 lines of declaring code—corresponding to 37 packages of task-calling instructions—from the Java SE API so that programmers already familiar with the Java language could easily work with the Android platform.

In 2010, Oracle Corporation bought Sun Microsystems, the original developer of Java SE, and brought suit in the Northern District of California for copyright infringement against Google, alleging literal copying of the declaring code in these 37 packages from the Java SE API, and nonliteral copying of the organizational structure of the API.[3] After a jury trial, the jury found copyright infringement, but deadlocked on Google's fair use defense.[4] However, the judge ruled that, because Google had written its own implementing code, the copied material was no more than a "system or method of operation," which could not be copyrighted.[5] The Federal Circuit reversed, holding that the API's declaring code and organizational structure could both be copyrighted, and remanded for further factual findings on the fair use defense.[6]

On remand, the District Court conducted a second trial, and the jury found that Google had proven fair use. On appeal, the Federal Circuit again reversed, concluding that, while it assumed factual questions in Google's favor, the ultimate determination of whether the facts met the requirements for fair use was a question of law.[7] The Federal Circuit concluded, as a matter of law, that Google's copying of the Java SE API declaring code was not fair use, holding that "[t]here is nothing fair about taking a copyrighted work verbatim and using it for the same purpose and function as the original in a competing platform."[8]

After the second Federal Circuit decision, the Supreme Court granted Google's petition for certiorari as to both the copyrightability of the Java SE API and Google's fair use defense.

THE SUPREME COURT'S DECISION

The Supreme Court considered two questions: (1) whether a fair use defense is a question of law, and (2) whether Google had satisfied the four-factor statutory test for fair use to prove its affirmative defense. "Given the rapidly changing technological, economic, and business-related circumstances" at issue in the case, the Court declined to reach a third question—whether the Java SE API was copyrightable in the first instance—as it was unnecessary to the disposition of the case.[9] Assuming that the code copied by Google was copyrightable, the Supreme Court reversed the Federal Circuit and held that, as a matter of law, Google's copying did in fact constitute a fair use.[10]

Whether There Is Fair Use Is a Legal Question. At the Supreme Court, Google argued that fair use is a factual question for the jury, subject to only limited appellate review to determine whether "substantial evidence" supported the verdict.[11] The Federal Circuit had held that fair use is a mixed question of law and fact, with the ultimate legal question being subject to *de novo* appellate review.[12] In its opinion, the Supreme Court endorsed the Federal Circuit's view that fair use is a mixed question of law and fact, but that the ultimate determination is a legal one subject to *de novo* appellate review.[13]

In reaching this conclusion, the Supreme Court explained that for a mixed question of fact and law, "a reviewing court should try to break such a question into its separate factual and legal parts" and that the ultimate "standard of review for a mixed question all depends—on whether answering it entails primarily legal or factual work."[14] The Supreme Court then determined that the fair use question "primarily involves legal work."[15] Thus, while deference is due to the jury's determinations on "subsidiary factual questions," the ultimate question remains a legal one.[16]

Application of the Four-Factor Fair Use Test. Having announced the appropriate standard of review, the Supreme Court turned to the application of the four factors for determining fair use set out in § 107 of the Copyright Act: (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. The Court noted that the provision's list of factors is "not exhaustive" (although it did not identify any other factors) and that "some factors may prove more important in some contexts than in others."[17]

The Supreme Court concluded that all four of the factors weighed in favor of finding that Google's copying constituted fair use.

"The Nature of the Copyrighted Work." Beginning with the second statutory factor, the Supreme Court distinguished between the uncopied implementing programs and the copied declaring code: The creativity involved in the implementing programs—the "'magic' practiced by an API developer when he or she worries 'about things like power management' for devices that 'run on a battery'"[18]—was different than the kind of creativity involved in the declaring code—which is designed to be "user-centered," intuitive, and easy.[19] The declaring code was "functional in nature" and "inherently bound together with uncopyrightable ideas."[20] Furthermore, "its value lies in its efforts to encourage programmers to learn and to use that system."[21] Ultimately, the Court found 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," and this statutory factor pointed towards finding fair use.[22]

"The Purpose and Character of the Use." This statutory factor focuses on whether the copier's use "adds something new, with a further purpose or different character, altering the copyrighted work with new

expression, meaning or message."[23] This factor is often described as asking whether the copying use is "transformative" of the original work. The Supreme Court reasoned that, although Google had precisely copied the portions of the Java SE API at issue, the analysis "must go further and examine the copying's more specifically described 'purpose[s]' and 'character.'"[24]

The Supreme Court agreed with Google that the copying at issue was transformative because it assisted in the creation of new products and was, thereby, "consistent with that creative 'progress' that is the basic constitutional objective of copyright itself."[25] Google copied only as much of the Java SE API as was needed for smartphone programs and only as much code as was needed to allow programmers familiar with the Java language to use it in the Android platform. Noting "the numerous ways in which reimplementing an interface can further the development of computer programs,"[26] the Court held that this factor, too, weighed in favor of finding fair use.

"The Amount and Substantiality of the Portion Used." The key question under this factor is the relative importance of the copied material to the original copyrighted work. The Court concluded that the inquiry should not primarily focus on how many lines of code Google did copy, but rather should "take into account the several million lines that Google did not copy," including the key implementing code.[27] The Court found that Google copied only the code needed to carry out its "legitimate objective": to "permit programmers to make use of their knowledge and experience using the Sun Java API when they wrote new programs for smartphones with the Android platform."[28] Again, the Court found, this factor weighed in favor of fair use.

Market Effects. The Supreme Court considered evidence of the competitive market in smartphones and concluded that this factor also weighs in favor of fair use. While more fact-bound than other factors, the Court nevertheless concluded that Sun's "uncertain" ability to compete in the smartphone marketplace, the fact that the sources of Sun's lost profitability as a result of the copying had "much to do with third parties' (say, programmers')" investments, rather than Sun's, and the potential for "creativity-related harms to the public" all favored a finding of fair use.[29]

Justice Thomas's Dissent. In a dissent joined by Justice Alito, Justice Thomas took issue with the majority's analysis, in particular that the majority assumed without deciding that the API code is protected by copyright. When viewed properly, the dissent argued, the Court's "fair-use analysis is wholly inconsistent with the substantial protection Congress gave to computer code."[30] The dissent suggested that, contrary to the majority's finding, declaring code is "closer to the core of copyright" than implementing code, because implementing code is not visible to developers.[31] The dissent pointed out that "the single most important element of fair use" is the effect upon the potential market for the copyrighted work, and that Google's copying "had a disastrous effect" on Oracle's ability to compete in the mobile operating system market, and reduced its ability to license Java in other markets.[32] The dissent also noted that if, in light of the Court's decision, "companies may now freely copy libraries of declaring code whenever it is more convenient than

writing their own, others will likely hesitate to spend the resources Oracle did to create intuitive, well-organized libraries that attract programmers and could compete with Android."[33]

IMPLICATIONS

The Supreme Court's decision not only upholds Google's use of the Java SE API declaring code in its Android platform, but may also indicate a broader discomfort with a strong form of copyright protection for computer code. The majority emphasized the downstream creativity enhancements that can come from greater ability to copy computer code, and downplayed the commercial impact of copying on a copyright holder's ability to exploit its intellectual property. While the Court focused on the ability of copiers to create new markets through transformative "reimplementation" of software, it did not address the dissent's observation that the decision may discourage other software developers from expending resources to create, for example, "intuitive, well-organized libraries that attract programmers." Nor did the Court consider the impact of its decision on the application of the fair use defense in other markets where copyright protection against limited copying may be important, such as publishing—although the Court stated that its decision did not overturn any of its prior fair use precedents.

The Supreme Court's decision is unlikely to settle other disputes related to copyright protection of computer code. By declining to address the question of the copyrightability of the API computer code at issue, and by electing a flexible many-factor balancing test for fair use rather than a more focused analysis, the Court rejected an approach that might have provided more predictive clarity. As a result, although the opinion and analysis will offer ammunition for those seeking to make fair use of copyrighted computer code, the approach taken by the Court provides less guidance than it might have on how other disputes regarding limited copying of software should be resolved in the future.

* * *

ENDNOTES

1 *Google LLC* v. *Oracle Am., Inc.*, No. 18-956 (S. Ct. Apr. 5, 2021).

2 *Id.* at 3–4 (quoting *Oracle Am., Inc.* v. *Google, Inc.*, 750 F.3d 1339, 1349 (Fed. Cir. 2014)).

3 Oracle also originally brought claims for patent infringement, which have since dropped out of the case. *See Google*, No. 18-956, slip op. at 9.

4 *See Oracle Am., Inc.* v. *Google Inc.*, 872 F. Supp. 2d 974, 976 (N.D. Cal. 2012).

5 *Id.* at 977 (citing 17 U.S.C § 102(b)).

6 *Oracle Am.*, 750 F.3d at 1354.

7 *Oracle Am., Inc.* v. *Google LLC*, 886 F.3d 1179, 1210 (Fed. Cir. 2018).

8 *Id.*

9 *See Google*, No. 18-956, slip op. at 15.

10 *Id.* at 1.

11 *Id.* at 18.

12 *Id.* at 19.

13 *Id.*

14 *Id.* (quotation omitted).

15 *Id.*

16 *Id.*

17 *Id.* at 14 (discussing 17 U.S.C. § 107).

18 *Id.* at 23 (quoting the testimony of a trial witness).

19 *Id.*

20 *Id.* at 23–24.

21 *Id.* at 24.

22 *Id.*

23 *Id.* (quotation omitted).

24 *Id.* at 25 (quoting 17 U.S.C. § 107(1)).

25 *Id.* (citation omitted).

26 *Id.* at 26.

27 *Id.* at 29.

28 *Id.* at 30.

29 *Id.* at 34–35.

30 *Google*, No. 18-956 (Thomas, J., dissenting), at 1.

31 *Id.* at 10 (quotation omitted).

32 *Id.* at 13.

33 *Id.* at 14.

ABOUT SULLIVAN & CROMWELL LLP

Sullivan & Cromwell LLP is a global law firm that advises on major domestic and cross-border M&A, finance, corporate and real estate transactions, significant litigation and corporate investigations, and complex restructuring, regulatory, tax and estate planning matters. Founded in 1879, Sullivan & Cromwell LLP has more than 875 lawyers on four continents, with four offices in the United States, including its headquarters in New York, four offices in Europe, two in Australia and three in Asia.

CONTACTING SULLIVAN & CROMWELL LLP

This publication is provided by Sullivan & Cromwell LLP as a service to clients and colleagues. The information contained in this publication should not be construed as legal advice. Questions regarding the matters discussed in this publication may be directed to any of our lawyers or to any Sullivan & Cromwell LLP lawyer with whom you have consulted in the past on similar matters. If you have not received this publication directly from us, you may obtain a copy of any past or future publications by sending an e-mail to SCPublications@sullcrom.com.