

Google v. Oracle: SCOTUS Sides with Google on Fair Use, But Is The Ruling Narrower Than It Seems?

April 19, 2021 – The TMCA

by Connor J. Hansen and Stefan Szpajda



Related People

- Connor J. Hansen

Related Capabilities

- Trademark, Copyright + Advertising
- Intellectual Property Litigation

On April 5, 2021, the Supreme Court issued its decision in *Google v. Oracle*, ruling 6-2 in Google’s favor on the issue of fair use. So ends a decade-plus battle between two tech giants that many viewed as having the potential to reshape how computer programs are written and licensed across the software industry. Google’s win—which is plainly indebted to the amici curiae who persuaded the Court of the policy rationale for a finding of fair use—thrilled those who saw a potential win by Oracle as an existential threat to settled norms in the software industry. Others have received the decision with skepticism, questioning how the Court could have blessed Google’s verbatim copying of Oracle’s code and struggling to reconcile the result with the plain text of the Copyright Act and established fair use precedent. Now, as the dust settles, those looking to *Google v. Oracle* for guidance on what they can and can’t do with proprietary software they don’t own must decide what, if anything, they can draw from its holding. The answer may prove elusive, and expensive.

We’ve followed this story since the Court first granted certiorari in 2019, with posts laying out the questions before the Justices, analyzing copyrightability and fair use, and summarizing oral argument. (For those new to the subject matter, these earlier posts will help you get up to speed on this complex case.) Now, in our final post in this series, we turn to the result.

Holding

Noting that *Google v. Oracle* does not “overturn or modify” earlier cases involving fair use, the Court held that Google’s re-implementation of a “user interface” (i.e., 11,500 lines of declaring code from 37 Java API libraries) owned by Oracle was a fair use of that material as a matter of law.

Justice Breyer wrote the majority opinion, joined by Justices Roberts, Sotomayor, Kagan,

Gorsuch, and Kavanaugh. Justice Thomas dissented, joined by Justice Alito. Justice Barrett took no part in the decision.

The Majority Opinion

Although there were two questions before the Court, the majority ignored the first, making no express ruling on whether the declaring code at issue is copyrightable. As discussed below, this omission drew sharp criticism from the dissent, and is likely to exacerbate uncertainty over how the holding in *Google v. Oracle* should be applied. That said, the Court implicitly sided with Oracle on this question as fair use only comes into play if the underlying work is subject to copyright; the majority also acknowledged in several places that computer programs, including the declaring code at issue here, are “subjects of copyright.” This acknowledgement, however, only makes the majority’s silence on the first question before it more conspicuous and puzzling.

Instead of addressing copyrightability under 17 U.S.C. §§ 101 and 102, the majority devotes its opinion to an analysis of the fair use factors enumerated under 17 U.S.C. § 107, ultimately ruling that each one favors Google.

Factor 2: “The Nature of the Copyright Work.” Although this factor is actually the second fair use factor, for “expository purposes” the majority considered it first.

Under this factor, the majority compared declaring code and implementing code, and identified the attributes of declaring code that distinguish it from implementing code and—per the majority—weaken its protection under the Copyright Act. For example:

- declaring code is “inextricably bound together” with the division of computing tasks that “no one claims is a proper subject of copyright;”
- declaring code is “inextricably bound up” with the idea of organizing tasks in a manner that is not copyrightable;
- declaring code is “inextricably bound up” with the use of commands, i.e., method calls, that Oracle did not claim were copyrightable; and
- declaring code is “inextricably bound up” with implementing code, which Google did not copy from Oracle.

Although these observations were all made in the context of fair use, they track the arguments Google raised in objecting to the copyrightability of declaring code under the first issue before the Court. This, again, puts a spotlight on the majority’s decision to ignore the first issue before it, and invites the question of why it didn’t simply make these observations in that context. (More on that in our discussion of the dissent.) Further elaborating on the distinction between declaring code and implementing code, the majority concludes that declaring code “embodies a different kind of creativity” in that Oracle’s code was written to attract programmers by making it easy to remember and use.

Putting these ideas together, the majority observed that although both declaring code and implementing code are “functional in nature,” declaring code is “inherently bound together” with (1) uncopyrightable ideas (task division and organization) and (2) others’

new creative expressions, like Google's Android implementing code. The majority thus ruled that the nature of declaring code weighs in favor of fair use.

Factor 1: "The Purpose and Character of the Use." Here the majority considered whether Google's use added "something new, with a further purpose or different character," altering Oracle's declaring code. First addressing traditional concepts derived from precedent to consider whether the use was "transformative" under this factor, the majority acknowledged that Google precisely copied the code at issue, and used it "in part" for the same purpose as Oracle. This would typically cut against Google, but the majority added that, in the context of computer programs, those cannot be reasons to rule against fair use because it would improperly limit the doctrine of fair use. Thus, the majority decided that Google's purpose was to use declaring code created for use in desktop and laptop computers and apply it to smartphones. Although the majority stopped short of ruling that all such re-implementation of declaring code will weigh in favor of fair use under this factor, it observed that the record demonstrates that re-implementing declaring code in this manner "can further the development of computer programs." In advancing this view, the majority cited the contributions of amici supporting Google, referencing briefs submitted by Copyright Scholars, Microsoft, Computer Scientists, R Street Institute, and American Antitrust Institute. The amici convinced the majority that Google's use was transformative under this factor.

The majority also considered the commerciality and good faith of Google's use, finding that Google's commercial use was not dispositive in light of its "inherently transformative" use, and expressed skepticism regarding the role Google's alleged bad faith should play in the analysis, choosing to give it no weight. The majority found this factor weighed in Google's favor.

Factor 3: "The Amount and Substantiality of the Portion Used." Google copied the declaring code for 37 packages of the Sun Java API, totaling approximately 11,500 lines of code. If considered in isolation, the majority acknowledged that would be a lot of copying. But the majority declined to consider it in isolation, and instead took into account the several million lines that Google did not copy in ruling that this factor weighed in Google's favor.

Factor 4: "Market Effects." Here, the majority declined to consider the market effects of Google's copying by looking at the revenue Oracle lost as a result. Instead, the majority stressed Oracle's poor position for success in the smartphone market, and weighed it against Oracle's claim that Google's copying harmed it. The majority also took into account whether Google's copying produced "public benefits" related to the Copyright Act's concern for the "creative production of new expression." Significantly, the majority noted that a weighing of public benefit may not always be relevant, "not even in the world of computer programs." But it nonetheless decided to weigh them here in determining the likely market effects of Google's re-implementation.

Somewhat confusingly, in ruling against Oracle on this factor the majority reasoned that the success of a computer program may initially be attributable to its expressive qualities, but that over time it can instead become valuable "because users, including

programmers, are just used to it.” Oracle had previously warned that a ruling in Google’s favor would effectively punish it for Java’s success. And although that may be a reductive summary of the decision overall, the majority’s reasoning here raises questions as to when the success of a copyrighted work shifts from its expressive qualities to the inertia arising from wide adoption, and why that should reduce its protection under the Copyright Act. Those wondering will find little comfort in the majority’s conclusion that “given programmers’ investment in learning the Sun Java API, to allow enforcement of Oracle’s copyright here would risk harm to the public.”

The Dissent

The dissent wasted no time before taking the majority to task for its failure to address the first question before the Court, i.e., whether declaring code is copyrightable. Noting that the majority “purports to assume, without deciding, that the code is protected,” the dissent concluded that the majority’s fair use analysis is “wholly inconsistent with the substantial protection Congress gave to computer code.” The implication here is that the majority sidestepped the first question because, had it fleshed out its position, the reasoning would have made its fair use analysis untenable. Per the dissent, “the majority purports to save for another day the question whether declaring code is copyrightable” because it “cannot square its fundamentally flawed fair-use analysis with a finding that declaring code is copyrightable.”

As to the nature of the work, the dissent rejected the majority’s attempt to distinguish implementing code from declaring code, noting that each observation made by the majority about implementing code is equally true of declaring code. The dissent added that “it makes no difference” that the value of declaring code depends on how much time third parties invest in learning it because “[m]any other copyrighted works depend on the same.” The dissent illustrates this point by analogy, explaining that although a Broadway musical script needs actors and singers to invest time learning and rehearsing it “a theater cannot copy a script—the rights to which are held by a smaller theater—simply because it wants to entice actors to switch theaters and because copying the script is more efficient than requiring the actors to learn a new one.”

The dissent also took a sharply different view of the market effects of Google’s copying, noting that whether or not Oracle could have built a smartphone on its own is only “half the picture” because Oracle could have licensed its software for use in Android. According to the dissent, Google’s copying destroyed that market for Oracle: “By copying Oracle’s work, Google decimated Oracle’s market and created a mobile operating system now in over 2.5 billion actively used devices, earning tens of billions of dollars every year. If these effects on Oracle’s potential market favor Google, something is very wrong with our fair use analysis.”

On the nature of Google’s use, the dissent accused the majority of conflating transformative use with derivative use, stressing that by the majority’s logic a movie studio’s unlicensed creation of a film based on a book would be transformative.

Finally, the dissent would find Google’s use substantial because it could serve as a

“market substitute” for the original.

At bottom, the dissent rejects the majority’s view that there is a sufficient difference between implementing code and declaring code to render the former more protectable than the latter. The result of the Court’s ruling, per the dissent, is that it is now “difficult to imagine any circumstance in which declaring code will remain protected by copyright.”

Standard of Review

Many observers were surprised when, in 2018, the Federal Circuit reversed the jury’s 2016 finding of fair use, and Google argued that the appellate court applied the wrong standard of review by failing to give appropriate deference to the jury’s findings of fact. Google went so far as to argue that the Federal Circuit violated the Seventh Amendment’s right to a trial by jury, which raised eyebrows when it spurred a request for supplemental briefing from the Court.

But on these procedural questions, both the majority and dissent agreed with Oracle that the Federal Circuit applied the correct standard when it treated fair use as a mixed question fact of law. The Court held that appellate courts should defer to the jury on findings of underlying fact, and then consider *de novo* whether those facts support fair use. Although it made no difference to the outcome here, the Court’s ruling on these questions injects more uncertainty into future litigation by making appeals from a jury’s findings on fair use more viable than they would be had the Court sided with Google on this issue.

Where Do We Go From Here?

It is tempting to view the Court’s ruling as a clear win for software developers who want to re-implement portions of proprietary computer programs without a license—i.e., much of the software industry. But the holding may prove narrower and more uncertain than it appears. That’s because any fair use analysis necessarily mixes questions of fact and law, and is often—by design—uncertain. Indeed, the majority in *Google v. Oracle* stressed the doctrine’s “flexible” approach to the “sometimes conflicting” aims of copyright law, and that “its application may well vary depending upon context.”

With the above in mind, consider the meandering path Google’s fair use defense took in this case. During the first trial in 2012, the jury deadlocked on fair use. In 2016, after a second trial on fair use, the jury found for Google. Then, in a rare move, the Federal Circuit reversed the jury’s finding and ruled that only one of four factors favored Google, with two of the four strongly or heavily favoring Oracle. Now, in 2021, the Supreme Court has ruled that the 2016 jury got it right, not only reversing the Federal Circuit but holding that every single fair use factor favored Google. The dissent, examining the same facts, found only one factor favored Google.

In addition, as illustrated in the chart below, the many amici who weighed in were split across factors. 39 amici addressed at least one fair use factor, but very few of the 39 addressed each of the four factors. The first factor was the most addressed factor, with

31 amici offering arguments relating to the purpose of the use, followed by the fourth factor with 26 amici addressing market effects. Although more amici argued in favor of Oracle for each of the fair use factors, the amici who argued for Google generally provided a deeper analysis of the issues relevant to fair use. This is because most Oracle-supporting amici who addressed fair use also addressed copyrightability, whereas many Google-supporting amici who addressed fair use devoted their entire briefs to fair use. Google-supporting amici also tended to focus on fewer factors, often just one or two factors, whereas Oracle-supporting amici tended to address three, if not all four, of the fair use factors. And the majority was clearly persuaded by Google-supporting amici, as it cited several of their briefs in its opinion.

Phase	Fair Use Factors			
	Purpose of Use	Nature of Work	Amount Used	Market Effect
Jury (2012)	Deadlock			
Jury (2016)	Google			
CAFC (2018)	Oracle (strongly)	Google	Neutral/Oracle	Oracle (heavily)
Amici (2020)	Google: 7/31 Oracle: 20/31 Neutral: 4/31	Google: 6/19 Oracle: 8/19 Neutral: 5/19	Google: 4/19 Oracle: 11/19 Neutral: 4/19	Google: 7/26 Oracle: 15/26 Neutral: 4/26
SCOTUS (2021)	Google	Google	Google	Google
Dissent (2021)	Oracle	Google	Oracle	Oracle

Now imagine you're a software developer who must decide whether *Google v. Oracle* clears the way for you to use another company's proprietary code. How do you determine whether the code at issue is more like implementing code, which the Court did not include in its ruling on fair use, or declaring code? And assuming you can determine that the code is like the declaring code here, do you look at the result—the Court ultimately found fair use, and resoundingly so—and find comfort to proceed without a license? Or does the expensive and uncertain path of a fair use defense make you hesitant to take on that risk? After all, like Google, you may prevail, but the victory may cost more in legal fees than a license would have. Or, you might lose outright because the highly factual analysis cuts against you. For example, a court might find that your use of the code did not result in a sufficiently successful product to warrant a finding of fair use under *Google v. Oracle*, or that the code's owner is better placed to compete in your market than Oracle was to compete in the smartphone market, and rule against you on that basis. Moreover, it should not be lost on anyone that Google spent many millions in legal fees, and the collective investment by the amici who filed briefs in support of Google was also significant. Query whether the same result would have been possible for a litigant without such resources to spare.

At best, in situations where the code in question is closely analogous to the declaring code at issue in *Google v. Oracle*, the Court's ruling may support an efficient resolution finding fair use. But the line between implementing code and declaring code in Java—not to mention its analogues in other programming languages—may prove difficult to draw. And in those situations, *Google v. Oracle* is unlikely to provide the sort of certainty that can lead to efficient resolutions of copyright disputes.

Conclusion

The Court ruled that, under the doctrine of fair use, Google was free to use 11,500 lines of code from 37 Java API libraries owned by Oracle when Google programmed its Android platform. For Google, this means it will not be liable for potentially billions of dollars in damages. What it means for everyone else remains to be seen, and in light of the fact-intensive and context-specific premises behind the Court's ruling *Google v. Oracle* may prove narrower than it first appears.