

Supreme Court Finds Google's Use of Oracle's Java Code in Android Operating System to Be Fair Use

April 7, 2021

On April 5, 2021, the Supreme Court of the United States held that Google's use of certain Java Application Programming Interfaces (API) in its Android operating system was not copyright infringement and instead constituted fair use of Oracle's Sun Java API because Google used "only what was needed to allow users to put their accrued talents to work in a new and transformative program." In its decision, the Supreme Court articulated important policy considerations underlying its decision, noting that, "given programmers' investment in learning the Sun Java API here would risk harm to the public. Given the costs and difficulties of producing alternative APIs with similar appeal to programmers, allowing enforcement here would make of the Sun Java API's declaring code a lock limiting the future creativity of new programs" and interfere with the basic objectives of copyright law. In sum, the Supreme Court relied on policy considerations relating to the ability of programmers to use existing code to support the interoperability of software, a common practice that many in the industry advocated as a practice necessary to sustain the feasibility of mobile computing.

This case spans nearly a decade of litigation between Oracle and Google. After negotiations broke down between Google and Oracle's predecessor to license the entire Java platform for development of Google's Android operating system, Google developed its own platform tailored to be used exclusively with Android smartphone technology. In developing the platform, Google wrote millions of lines of unique code, but also copied 11,500 lines of code from the Java SE API. The API operated to identify and group tasks and to call up prewritten software to carry out those tasks. Google's own code operated to actually carry out the called-up task. After Oracle acquired the owner of the Java SE API and its corresponding copyrights, Oracle sued Google for copyright infringement.

Importantly, the Supreme Court did not address the question of whether the API was eligible for copyright protection. Instead, for purposes of this case, the Supreme Court assumed the API was copyrightable, and held that Google's use thereof was a fair use and did not violate copyright law. Writing for the majority, Justice Breyer examined the four guiding principles identified in the Copyright Act's fair use provision: (1) the nature of the copyrighted work; (2) the purpose and character of the use; (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. The Court found that all four factors weighed in favor of fair use.

The Supreme Court defined the nature of the work as a user interface, thus, the API was “inextricably bound up” with an organizational system—which is not copyrightable—and with an implementation system—which is copyrightable (but was not copied in this case). The Court held that overall, the program is further from “the core of copyright” than most computer programs and that this weighed in favor of fair use.

The Supreme Court held that Google’s use of the program was transformative in that its purpose was to create new products through the Android platform. The Court found this use was consistent with the constitutional objective of the Copyright Act to promote creative progress, which also weighed in favor of fair use.

With respect to the amount and substantiality of the portion of code used in relation to the copyrighted work as a whole, the Court did not consider the 11,500 lines of code as a single complete work but rather as a small part of a larger program consisting of several million lines of code—the rest of which Google did not copy. The court found that this factor weighed in favor of fair use.

Finally, in considering the effect of the use on the market, the record showed both that Google’s new smartphone platform was not a market equivalent of Java SE *and* that Java SE’s copyright holder would benefit from the reimplementations of its user interface into a different market. The Supreme Court held that enforcing copyrights based on the facts in this case would cause “creativity-related harms to the public” favoring fair use.

The Court’s detailed analysis of copyright fair use in the context of software programming provides much-needed clarification for software developers that engage in the common practice of using and reusing interfaces written by others. The Supreme Court, however, left open the question of whether such code is copyrightable as a matter of law. In a scathing dissent, Justice Thomas wrote that this decision “eviscerates copyright” and that the only reason the majority chose not to address the question of copyrightability was “because the majority cannot square its fundamentally flawed fair-use analysis with a finding that declaring code is copyrightable.” Whether such programming code is subject to copyright protection as a matter of law will likely be the subject of future debate and lawsuits alongside the ever-developing landscape of software programming.