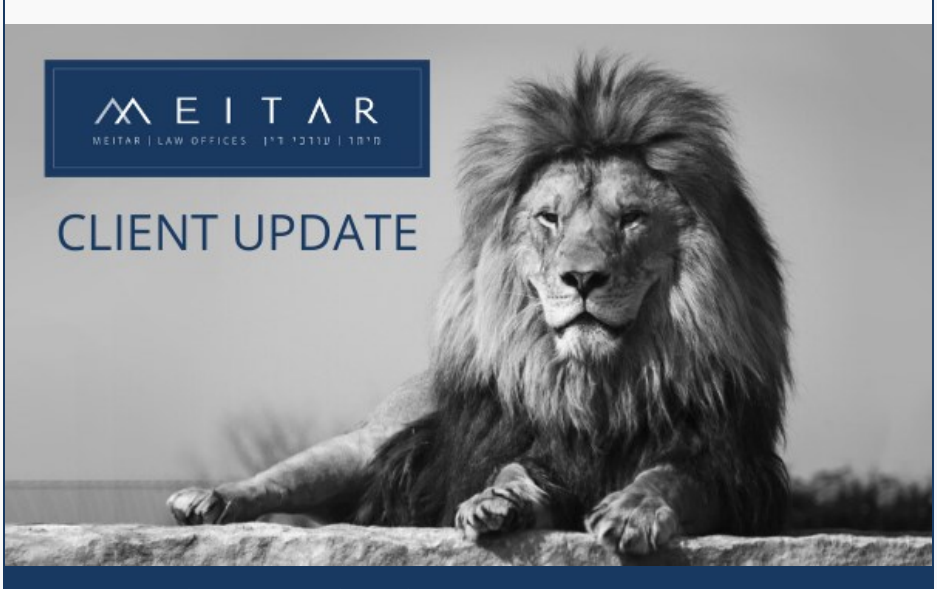
View this email in your browser



Google v. Oracle: The US Supreme Court gives significant guidance on "fair use"

May 3, 2021

In the most important case of what constitutes permitted "fair use" of a copyrighted work in more than a quarter century, the United States Supreme Court held, 6-2, that Google's copying of parts of Oracle's Java SE platform for Google's Android mobile phone platform was fair use and that Google was therefore not liable for copyright infringement. This case is significant because it gives guidance as to how our clients should analyze issues of fair use, particularly in the context of using an Application Programming Interface (API) of other computer programs. Although Justice Breyer, writing for the majority, stated that "we do not overturn or modify our earlier cases involving fair use," the opinion is notable in its emphasis on the centrality of the concept of "transformative use" in determining whether the use is fair, and in the Court's focus on the public policy issues of innovation and the impact of its decision on creativity, rather than applying a more narrow reading of the statute. Moreover, given that the Israeli statutory provision on fair use is based almost word-for-word on the U.S. provision, and that Israeli courts often look to U.S. decisions, the case merits note for its potential indirect impact on the development of fair use in Israel.

Factual Background

In 2005, Google acquired Android and used it to develop a software platform for mobile devices like smartphones. Google wished to use the Java programming language, popular among software developers, and discussed licensing it from Sun Microsystems, the developer of Java. The parties did not come to an agreement. Nevertheless, Google copied 37 libraries from the Java API, comprising about 11,500 lines of code. In 2010, Oracle (having purchased Sun Microsystems) sued for copyright infringement.

At the trial court, the jury found that Oracle's API was indeed protected by copyright but that Google's use was fair use. On appeal, the Federal Circuit reversed, holding that Google's use of the Java API did not constitute fair use. It stated: "there 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." It is from this decision that Google appealed to the Supreme Court.

What did Google actually copy?

It copied the declaring code of 37 packages in the API. The API allows programmers to use prewritten code to build certain functions in their own programs, rather than write their own code to perform these functions. Through the API, the programmer can draw on a large library of prewritten code to carry out various tasks (like answering the question: "which number is higher?" or sorting thousands of numbers from high to low). In the Java API, each task is a "method"; similar methods are grouped into "classes," and similar classes are grouped into "packages".

For each task, there is "implementing code," which tells the computer how to execute the particular task selected. Google wrote all of its own implementing code, so there was no claim of copyright infringement on the implementing code.

How do you tell the computer which implementing code to choose? You do so through a "method call." And how does the method call (which a programmer types) locate the implementing code? Through the "declaring code." Without the declaring code, the method calls entered by the programmer would not call up the implementing code.

Google copied these shortcuts, the declaring code. Was that fair use ?

The Majority Opinion

The 6-2 majority held that it was fair use. In so holding, it emphasized that copyright should not be used as a tool to stifle innovation, and the monopoly which copyright grants should not grant more economic power than is necessary to achieve the incentive to create.

Another major theme throughout the opinion is that computer programs differ significantly from other copyrighted works, like books, films and music, in that they almost always serve a functional purpose, and therefore are entitled to more limited protection—or at least the declaring code of an API is entitled to less protection.

Guided by that philosophy, the Court then applied the four factors set out in the Copyright Act for determining whether it was a fair use. The Court held that all four favored fair use.

The Nature of the Copyrighted Work

This factor evaluates how "creative" the work is—the more creative, the more protection it deserves.

The Court reiterated what declaring code does—it labels the tasks in the API and organizes those tasks (methods) into packages and classes—which the Court referred to as "file cabinets, drawers, and files".

The Court therefore believed that the declaring code was less creative that other kinds of software, and that it is bound up with uncopyrightable ideas (general task division and organization) and new creative expression (Android's code). In addition, the Court believed that the value was not inherently in the creativity of the code itself, but rather derived mostly by how much third party programmers invest of their own time and effort to learn the API's system. It is interesting and, in my view, not necessarily obvious, how much weight and deference the Court gave to the programmers being the ones who did the creative work and that they should not be unduly burdened. Accordingly, this factor favors fair use.

The Purpose and Character of the Use

This factor looks at whether the copier's use fulfills the objective of copyright law to stimulate creativity. If the copier's use is "transformative" – that is, if it adds something new and important – courts are likely to find fair use. Although the Court has stated that all four factors needed to weighed together, in many ways, transformative use is the key fair use issue.

The Court described how Google's use of the Java API sought to create new products, and to expand the use and usefulness of Android phones. The Android platform was, in the Court's view, a highly creative and innovative tool for the smartphone environment. Therefore, if Google's programmers needed to use the Java API to create a new platform, this was consistent with the purpose of copyright law, which is not to reward authors, but, as set forth in the US Constitution, to "promote the progress of science and the useful arts".

The Court listed the ways that Google's "reimplementing" of the API interface in a new system furthers the development of computer programs, finding that such reimplementation is:

- necessary for different programs to speak to each other; and
- necessary for developers to use their acquired skills. Without it, they need to learn a whole new language to use API labels.

The Court also pointed out that reuse of APIs is common.

The court held that Google's use was transformative, and that this factor favors fair use. Interestingly, it gave little weight to the element of the copier's good faith, quoting a seminal Harvard Law Review article on fair use: "Copyright is not a privilege reserved for the well-behaved".

The Amount and Substantiality of the Portion Used

This factor looks at the amount taken of the entire work, judged both qualitatively and quantitatively. Even if the amount taken is not significant quantitatively, if it is the "heart of the creative work", this would weigh against a finding of fair use.

The Court agreed that if you viewed the declaring code in isolation, Google's copying of 11,500 lines of code was significant. Because the Court determined that the declaring code was inextricably linked to the implementing code, however, the total code amounted to 2.86 million lines. Looked at this way, the declaring code was not substantial quantitatively, as it constituted only 0.4% of the total code. Since the Court viewed the implementing code as the core of the creativity, the fact that Google did not copy the implementing code but rather wrote its own code for a valid, highly transformative purpose (its smartphone computing environment) meant that this factor also favored fair use.

Market Effects

This final factor looks at the impact that the copying will have on the market or potential market of the original copied work. Where the copied part is a substitute for the original, there is less likely to be fair use.

The Court believed that there may not have actually been such a large loss here by Oracle, as its primary market for Java SE was laptops and desktops, and not smartphones. In addition, the Court recounted the evidence that Oracle had failed in its attempts to enter the smartphone computing market and was not well-positioned to do so. In short, since Android was not a market substitute for Java's software, the Court found that this factor also favored fair use. Furthermore, there was evidence that, on the contrary, Oracle stood to benefit from the adoption of Java to the Android platform and from the expansion of the Java network of developers. Taking a broader view of the overall potential market consequences, the Court was concerned that, given programmers' investment in learning the Java API, if the Court permitted Oracle to enforce its copyright here, it could prevent the creation of new programs, given the cost and difficulty of producing alternative APIs with similar appeal to programmers.

Takeaways and Impact for Clients

This Court decision is dramatic in its emphasis of the public policy issues behind the copyright law and the fair use provisions. It interpreted these very broadly, with an emphasis on whether the use by the copier (Google) was to create a new, creative and transformative product. If so, the use is fair. Focusing on that question should now be the first examination in any fair use inquiry.

The Court also viewed the transformative use through a wider lens than in the past. Instead of focusing on whether the actual use of the copied work was transformative, the Court took a much broader view--if the copying was part of a process which resulted in a significant creative work being developed, this copying would be considered to be transformative.

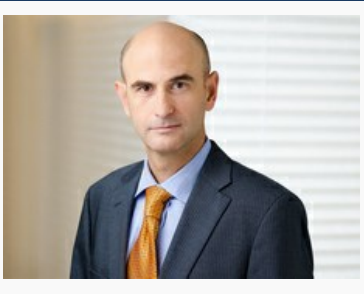
The Court was also concerned that a narrow reading of fair use for APIs of computer programs—or at least the declaring code therein—would grant a quasi-monopoly to Oracle and stifle innovation. This is particularly because it would burden software developers who had invested so much in learning the Java API.

Following this case, in the software context in particular, code that is seen to be functional is likely to have less protection if copied, particularly if the end use by the copier is creative. By permitting the reuse of functional code as a building block for create new, transformative products, this case can be seen as advancing the ability to leverage the knowledge of developers for collaboration, and to create interoperable products that build on well-known APIs.

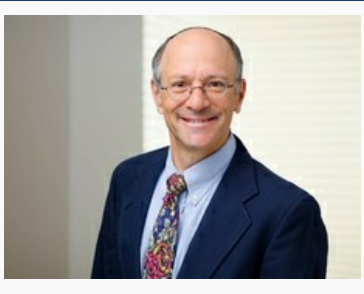
Israel has a virtually identical fair use statute to that of the United States. In stark contrast to this case, however, Israeli courts have repeatedly emphasized the importance of "good faith" as a component of the fair use analysis. It will be interesting to see what, if any, impact this case will have on Israeli fair use jurisprudence.

We will be closely following future cases in the United States and Israel to also see if the wide-ranging, favorable views of fair use in this case spill over into copyright cases not involving software.

Contact Information



Dr. Yoav Oestreicher, Partner
+972-3-6103980
yoavo@meitar.com



David Mirchin, Partner
+972-3-6103199
dmirchin@meitar.com

For additional information about our firm's Technology group, click here

To join our newsletter click here