LAW360

Portfolio Media. Inc. | 860 Broadway, 6th Floor | New York, NY 10003 | www.law360.com
Phone: +1 646 783 7100 | Fax: +1 646 783 7161 | customerservice@law360.com

With High Court Mum On Java Copyrights, Is Innovation Safe?

Law360, New York (July 1, 2015, 6:18 PM ET) -- Earlier this week, the U.S. Supreme Court decided not to review the Federal Circuit ruling in Google Inc. v. Oracle America Inc. We asked our Voices of the Bar panel to share some thoughts on the implications for software innovation and interoperability. Here's what they said ...

Question: The Supreme Court has declined to hear Google v. Oracle. Many critics of the underlying Federal Circuit ruling worried that, if left in place, it would stifle innovation and harm interoperability. Do you think these concerns are valid?

Steven Wong, The Home Depot Inc.
I think the concerns are overstated. It is well-established that software code is protectable by copyright as a work of authorship. In this regard, it is not unreasonable for a court to find that application programming interfaces (APIs), including their structure, sequence and organization, are entitled to protection under the Copyright Act. Copyright still does not protect the underlying function, and programmers are free to create their own code that performs the same function, with such code having its own structure, sequence and organization.

Google v. Oracle does not change copyright law, but merely clarifies that APIs are protectable. Like the patent bar has done with Alice, the copyright bar will similarly adapt to Google v. Oracle. Going forward, developers will need to consider the increased risks of incorporating APIs without permission. Because APIs are typically freely given away under license, it should mostly be a non-issue except in limited instances where the user chooses not to accept the licensing terms and still decides to copy the API (as was the case in Google v. Oracle). Even in such instances, the developers will still be entitled to present defenses such as fair use.

It will be interesting to follow the district court on remand as it decides the fair use question. A finding of fair use may substantially limit the practical effect of the Federal Circuit’s ruling.

Oren J. Warshavsky, BakerHostetler
In a vacuum, the Federal Circuit’s holding could well disrupt interoperability. APIs are the specifications permitting programs to communicate and operate together across a single or multiple platforms, and thus are critical to, among other things, local networks, mobile computing, cloud computing and the Internet of Things. APIs are openly used throughout the computer industry; developers write programs for one platform and then use APIs to make the application operable across other platforms. The industry trend has been toward universal applications that are “platform agnostic” — the same application is written for a desktop computer, iPhone, Android phone, etc. If all APIs are protected copyrights, and enforced by the copyright owners, the Federal Circuit’s ruling would require developers

to rewrite applications for each platform and interoperability certainly would be chilled.

But the Federal Circuit was interpreting Ninth Circuit law, and many predict the Ninth Circuit — and other circuits — will decline to follow this ruling. Moreover, the decision can be limited because it focuses on the creative choices when these APIs were created; other APIs are created differently and many result from an iterative, purely utilitarian (and therefore not creative) process. Finally, the ruling specifically notes that interoperability will be considered in the fair use analysis in this case — which convinced the solicitor general that the Supreme Court should refrain from taking the case. The Federal Circuit's decision clearly deals a blow to certainty — for now there is no longer a bright line copyrightability test — but its impact on interoperability and innovation may not be felt as significantly as some fear.

Jeff Van Hoosear, Knobbe Martens Olson & Bear LLP

An indisputable impact is that businesses and program developers will face uncertainty over the effect of their use of third-party APIs in software programs without obtaining authorization or a license. However, this uncertainty may not be the case outside of the U.S., as the European Court of Justice has reached a different conclusion as to copyright under European law. Arguably this difference is due to a policy that the European Union made long ago on the issue of software operability and copyright and not under copyright law. Looking at Oracle's APIs only as to U.S. copyright law (and not from any policy concerns), if the APIs meet the necessary criteria — which would basically be "originality" and a "modicum of creativity" — then copyright should apply. If the software industry has operated under the assumption that the functionality of APIs prevented copyright rights, then it must reconsider how it operates depending upon the final conclusion in this case. That said, it is important to remember that the case is not over, and will return back to the district court for a review of the fair use defense that Google has presented.

David L. Suter, Harness Dickey & Pierce PLC

While it's dangerous to read much into the Supreme Court's denial of cert in any given case, its refusal to review Google v. Oracle suggests the decision is neither so flawed nor so far reaching as many commentators suggest. The case will doubtless impact software developers looking to use Java API declaring code outside of Oracle's licensing scheme. And, the Federal Circuit's analysis will doubtless be cited regarding the merger and interoperability aspects of software copyrightability and infringement. Moreover, while applying Ninth Circuit law, its analysis will likely carry significant weight in software copyright litigation in other circuits, bolstered by the denial of cert.

All that said, it's difficult to say that Google v. Oracle will unduly stifle competition, perhaps unless one's perspective is that any successful assertion of intellectual property will constrain competition. Clearly this case will constrain those who develop software using Java APIs. On the other hand, the successful assertion of copyright in this case may prompt innovation to develop alternatives to the specific code that was found to be infringed. Such "design around" efforts may be difficult, perhaps even impossible as some have warned. But, the commercial success of the software should not, in and of itself, preclude its ability to be protected. Rather, as noted by the Federal Circuit, the ability to copyright software turns on the facts as they existed at the time it was created — not at the time of infringement.

To turn a phrase from the Federal Circuit's decision, the "devil is in the details" as to how the facts and analysis in Google v. Oracle will impact future cases. But, given the constraints on patentability imposed by the Supreme Court in Alice v. CLS, software developers may well look to copyright as the answer, bolstered by Google v Oracle.

Andrew W. Stroud, Hanson Bridgett LLP

The founding fathers included the Copyright Clause in the Constitution because they believed that securing legal protection for authors and inventors was a means of fostering innovation. This wisdom has been proven correct time and again. The age of invention in which we now live is fueled by the fact that innovators can capitalize on their creativity. The Google case is entirely consistent with this overriding principle. Copyright protection was awarded to the creators of a software program in order to reward them for their innovation. Innovation is rewarded not stifled.

Indeed, the reason Google lost was because they were not innovative enough. Google copied Oracle's declaring code verbatim rather than developing their own, although they could have easily done so. While this may have been more efficient, it was certainly not more innovative. In addition, it must be remembered that the case is not over yet. Google's fair use defense must still be tried. If Google prevails on this defense then concerns about the future impact of the decision will be greatly alleviated.

Finally, the best answer to issues of interoperability continues to be the market. If software companies want developers to be able to build off their systems, then they will provide them the means for doing so. If they do not, then interoperability will be a problem but it will be a problem for them more than anyone else as their system becomes isolated. The Copyright Act provides innovators with great power to control how their works are used. We should not alter or amend the act simply because we believe a few may abuse that power.

Robert Stoll, Drinker Biddle & Reath LLP

Courts have determined it is fair use to permit limited copying for interoperability when it is the only way to gain access to necessary fundamental elements and basic ideas in a computer program. Here it appears that Google copied a significant amount of the Oracle software. It is not clear that the amount copied was limited to that necessary for interoperability and the lower court has already acknowledged that there was nothing in the rules of the Java language that required Google to replicate the same language. It also seems that the purpose of the copying was not for interoperability but for making it easier for developers to write Android programs in Java.

It appears that Google used Oracle's Java API for commercial purposes. No standard open-source Java license that would have returned improvements to the Java open-source community was granted and Android is now inoperable with standard Java.

This case is about copyrighted works, which provide billions to domestic economic growth in the United States and are also a large contributor to job creation. Software has long been held to be copyrightable subject matter and much effort goes into the development of software. As with any copyrighted work, those seeking to use another's intellectual property for their own business purposes should take a license.

Mark Scarsi, Milbank Tweed Hadley & McCloy LLP

As a former software engineer turned lawyer, I'm a little torn by the Federal Circuit decision. While the ruling may have some impact on interoperability, it probably goes too far to say that it stifles innovation. If the Federal Circuit is right on the facts that the APIs at issue contain creative expression and are not purely functional, competent software engineers should, in theory, be able to create functionally equivalent APIs from scratch using different expression. But as every good software engineer knows, rewriting code that already works is — technically speaking — a horrible idea. It can lead to a host of unintended consequences. This result certainly highlights the benefits that can be recaptured if the industry were able to adopt an open source set of APIs for many common and basic functions. In the meantime, it will be important for technology litigators to bone up on their software knowledge so they can better understand (and explain) the traditional idea/expression copyright dichotomy as it pertains to software code.

Michael P. Sandonato, Fitzpatrick Cella Harper & Scinto

What has always been most interesting to me about the Federal Circuit decision that the Supreme Court has declined to review is its rejection of Google's policy-based arguments. In rejecting those arguments, the Federal Circuit made clear that patent protection and copyright protection for computer software are not mutually exclusive, and may comfortably co-exist. On the other hand, the Supreme Court's decision in Alice Corp v. CLS Bank and the jurisprudence that followed have seriously hampered a software developer's ability to protect their creations effectively through patents. One could argue that the Supreme Court's decision to leave Google v. Oracle untouched may actually promote innovation, in the sense that Google offers software developers a meaningful alternative to patent protection, in an era where patents on software have become increasingly difficult to enforce.

Gary Morris, Morris & Kamlay LLP

Interoperability is the watchword for software meant to run on connected devices, which means more and more devices in our increasingly networked world. So it's notable when courts issue decisions that can have wide-ranging impacts on innovations that make it easier for one device to communicate with another. Oracle v. Google involves Java and Android, two important technologies that enable devices to communicate with each other. After trying in good faith to reach a license agreement for Java with Sun Microsystems (later acquired by Oracle) in 2005, Google authored its own APIs for Android. The success of Android is undeniable. Today, over a billion devices around the world use the Android operating system. To make things easier for programmers working on interoperability, Google gave a few of its Android APIs the same names as corresponding Java APIs. The content of those Android APIs was different than what was in Java. As district court Judge Donald Alsop noted, 97 percent of the lines in the accused Android APIs are original to Google. The remaining 3 percent relate to declaration lines in the APIs that are based on the names and are necessary for the APIs to function. Judge Alsop held that the names and declaration lines in Java APIs are freely replicable under the names doctrine (names and short phrases are not copyrightable) and mergers doctrine (expression inseparable from its underlying idea is not copyrightable.) The Federal Circuit reversed and remanded to the district court to determine if Google is entitled to prevail on an affirmative defense of fair use. Given that the decision reaches far beyond the parties in suit, it's disappointing that the Supreme Court did not grant cert. The outcome could affect the pace of innovation in tying together devices in applications as diverse as home automation, manufacturing and telecommunications.

Patricia Martone, Law Office of Patricia A. Martone PC

No new law was created by the Federal Circuit in Google v. Oracle, and no barriers to innovation and interoperability were newly erected. The Federal Circuit applied long-established Ninth Circuit copyright law in holding that the declaring code and the structure, sequence and organization of Oracle API software packages were entitled to copyright protection.

The complaints about erecting new barriers arise out of the Federal Circuit's rejection of Google's argument that section 102(b) of the Copyright Act removes protection available to original works if the works also have a functional component. The Federal Circuit, in rejecting Google's argument, held that the court's role is to find the expressive aspects of the works and then separate out the aspects that cannot be protected, utilizing the Ninth Circuit's "abstraction-filtration-comparison" test. "This test [initially formulated by the Second Circuit] rejects the notion that anything that performs a function is necessarily uncopyrightable."

In its unsuccessful petition for certiorari, Google argued that the Federal Circuit should have ignored Ninth Circuit law and followed Lotus v. Borland, which holds that works with a functional component cannot be entitled to any copyright protection. Google stated that if the Federal Circuit decision was the "law at the inception of the Internet age," "vast amounts of technological development" could have been blocked. But the fact is that the law which the Federal Circuit followed has been good Ninth Circuit law since 1992. That law predates Lotus, widespread use of the Internet, and the founding of Google. The terrible impact predicted by Google did not happen. The United States is the acknowledged worldwide leader in software development, and the epicenter of this work is geographically located in the Ninth Circuit.

Floyd A. Mandell, Katten Muchin Rosenman LLP

APIs are an important tool in modern software development. They are of great assistance to third-party developers to expand upon existing technology, and the ability to freely use APIs has helped create competition and innovation in the software industry. To the same extent that there has been a push to limit patent protection, the same public policy goals could apply to overbroad protection in the copyright area. Unlike trademark law, where the principle goal is protecting the public from confusion, the argument is whether the public interest is really best served by restrictions in the use of APIs via copyright law. The concerns about hindrance on innovation and interoperability are valid, but the question is whether those concerns are outweighed by the interest in protecting the copyright owner and its investment, and to what extent a fair use argument should be applied.

Richard Z. Lehv, Fross Zelnick Lehrman & Zissu PC

This cases illustrates the "idea/expression" dichotomy. Section 102(a) of the Copyright Law says that copyright protection extends to "original works of authorship," including "literary works." It is well-settled that a computer program can be copyrightable as a "literary work." However, Section 102(b) of the Copyright Law says, "In no case does copyright protection for an original work of authorship extend to any idea, procedure ... [or] method of operation ... regardless of the form in which it is described, explained, illustrated, or embodied in such work."

Since a computer program can be said to be a "method of operation," given that it instructs the computer how to operate, Sections 102(a) and 102(b) are seemingly in conflict. The courts have resolved such conflicts by making a distinction between an idea (unprotectable) and the author's expression of the idea (protectable).

In addition, the fair use defense in Section 107 of the Copyright Law provides limits on the scope of protection for a copyrighted work.

In this case, the Supreme Court invited the solicitor general to file a brief "expressing the views of the United States." I agree with the views of the solicitor general, who pointed out that no final decision has been reached in the litigation: the Federal Circuit had remanded the case to the district court for further consideration of the fair use defense. The solicitor general recommends that the Supreme Court not hear the case before a final decision has been made, for the following reason:

“Although petitioner has raised important concerns about the effects that enforcing respondent’s copyright could have on software development, those concerns are better addressed through petitioner’s fair use defense, which will be considered on remand.”

Dr. Scott Kamholz, Foley Hoag LLP

I might have had some concern if the Supreme Court’s denial of cert. had been the end of Google v. Oracle. But it is not. The Supreme Court rightly declined to consider Google’s functionality defense and allowed a remand to the district court for consideration of Google’s fair use defense. After what will likely be a complex and expensive analysis, the case well may return to the Supreme Court on the fair use defense. That will be the appropriate time to consider the applicability of a fair use defense to declaring code generally and the stifling effects, if any, of the Java API copyrights in particular. It may be that Google can show its use is transformative, perhaps because it does not supplant the original purpose of the Java API, but rather draws developers to the Android platform instead of platforms that don’t mimic the Java API. In that case the net effect on innovation might be small. Google may also be able to show that its replication of the declaring code did not negatively impact Oracle’s market, but rather stimulated developers to develop apps they otherwise might not have. The facts developed on remand will govern the outcome, and the case will continue to attract significant attention.

Mark L. Hogge, Dentons

Innovation will be stimulated by upholding the copyright on Oracle's declaring code. The 7,000 lines of copied code meet the statutory requirements for a copyright so the author(s)/assignee(s) should have the exclusive right to control the property that their hard-earned capital and efforts created and deployed. Interoperability will not and cannot be harmed because either Google or the enabling programmers will draft from scratch something *new*. That something *new* will be an innovation and will take interoperability to the next level. We are all about disruption and reaching the new level in our system. Besides, Google still has its day in court on fair use.

Krish Gupta, EMC Corp.

EMC filed an amicus brief with the Federal Circuit before it decided the Oracle v. Google case urging the court to hold that the district court’s copyrightability analysis was erroneous. Contrary to the concerns of critics, a blanket rule that APIs are not within the scope of copyright protection would have impeded, not fostered, innovation. There is a difference between the question of whether copyright in a software product has been infringed and the question of what elements of software can be copyrighted in the first place.

Software platforms, like operating systems, provide “shared services” to third-party code through the use of APIs. Without copyright protection for the APIs, software developers would be hampered in their ability to restrict the use of their APIs in ways that could allow competitors to benefit for free or could harm the integrity of their software platforms. These issues would limit the willingness of software developers to publish their APIs, ultimately reducing interoperability and therefore innovation.

The availability of copyright protection for APIs solves these issues. Competitors cannot freeload off of the hard work of software platform developers, incentivizing such companies to put in the development effort needed to create first-class products. And developers of software platforms can license the use of the APIs to third parties, imposing conditions requiring the third parties to use the APIs only in intended, safe ways. As a result, developers will be more willing to publish their APIs, leading to the development of more interoperable software products.

Existing legal protections, including fair use, are regularly invoked by courts to find that it is permissible, particularly for interoperability purposes, to reuse significant portions of copyrighted works of computer software to develop products that offer better user experience.

Herbert D. Hart III, McAndrews Held & Malloy

Nothing in the Federal Circuit's ruling in Google v Oracle will stifle innovation; indeed, it will foster it. Though there is no question that there's a benefit to system interoperability, that alone is no justification for giving later developers a free pass to use any copyrighted software that may be useful in achieving it. The Constitutionally provided exclusivity to authors (copyright owners) is a key incentive to the creation of such works as the software we all use on a daily basis.

In the patent context, the need for interoperability is dealt with under the law governing what are determined to be standards-essential patents. As the name implies, such a patent claims technology needed for compliance with industry standards in such areas as WiFi. But a determination that a patent is a standards-essential one does not relieve a user of the obligation to pay royalty to the patent owner; it simply provides a framework for the licensing of such a patent.

Naomi Jane Gray, Harvey Siskind LLP

Despite the shrieks of rage and despair heard around Silicon Valley following the Supreme Court's denial of certiorari, predictions that the sky is falling on software innovation are premature. For starters, the case has been remanded to the district court to determine whether Google's copying of the declaring code of certain Java APIs constitutes a fair use. Thus, Google may yet prevail, paving the way for others.

Commentators have correctly noted that, from a defendant's standpoint, relying on fair use is significantly less advantageous than being able to argue that the work being copied is not protectable. Litigating fair use — typically decided on summary judgment or at trial — is risky and expensive, particularly in a software case involving expert testimony. And fair use is notoriously complex and unpredictable. The doctrine's reach, however, has greatly expanded in the last decade, as courts have broadly interpreted what uses are "transformative" — serving a different purpose than the original — and thus fair. And courts have begun to show greater willingness to find fair use on motions to dismiss, which, where successful, drastically reduces the cost of litigation.

More importantly, the Federal Circuit's opinion is unlikely to be the last word in this area. The Federal Circuit has exclusive jurisdiction over patent appeals. It heard Oracle v. Google because Oracle's complaint asserted patent claims, though those claims did not figure in the appeal. Because the case originated in California, the Federal Circuit was constrained to apply Ninth Circuit law to the copyright issues. The Federal Circuit's opinion, though maybe persuasive, is not binding on the Ninth Circuit. Thus, the Ninth Circuit is free to reach a different conclusion should it be presented with a similar case — as are other appellate courts. One thing is certain: more litigation is coming.

D. Bartley Eppenauer, Shook Hardy & Bacon LLP

Far from stifling innovation or harming interoperability, the Supreme Court's certiorari denial in Google v. Oracle and the Federal Circuit's ruling in the case simply confirm the state of software copyright protection that has been in existence for over 30 years during the incredible growth of the software industry. It simply lacks all credibility to assert that an industry that contributes several hundred billion USD annually to GDP suffers from a stifling effect to innovation. Moreover, during this timeframe, the fair use doctrine has accommodated widespread interoperability and significant reverse engineering and other re-use of computer code. Now the lower courts will have the opportunity to determine the applicability of the fair use doctrine in this case.

Copyright protects the creativity and design in software code, and the structure, sequence and organization (SSO) of software code does and should, in many instances, enjoy copyright protection. Software does not lose copyright protection simply because it has functional aspects, as the lower court in this case concluded. The degree of protection may vary in any instance of copying, depending upon whether code is literally copied or whether non-literal elements are infringed. And the Federal Circuit decision in this case recognizes a reasonable threshold for originality when assessing non-literal elements of software code.

To the extent that interoperability is at issue, the case law has made clear that interoperability arises as a question of infringement, and not of copyrightability. Without question, providing copyright protection for SSO of software does not prevent developers from making software interoperable. In cases where substantial non-literal copying is made, fair use can provide a reasonable defense, providing the factors for invoking fair use are met. As such, claims that the Federal Circuit's ruling in this case will threaten innovation or interoperability in the software industry are simply unfounded.

Chuck Ebertin, Intellectual Ventures LLC

The copyrightability of low-level computer code has been accepted for nearly 30 years. Google v. Oracle does not represent a fundamental change in the law of copyrightability.

The relevant Java API packages were original in the sense of copyrightability — they possessed at least some minimal degree of creativity in the naming of 7,000 different methods/classes. The originality requirement for copyrightability of literary works (including software) under section 102(a) of the Copyright Act is still understood to be a low threshold.

Defendants arguing for the absence of copyrightability in a software copyright case typically fail when it is established that the same functions performed by the accused code could have been achieved with differently written code. The evidence in this case illustrated that Google could have generated its own naming conventions for the methods and classes to achieve the same result. In short, the evidence illustrated that expression, and not mere functionality, was copied.

Denial of certiorari in this case should have no impact on innovation or interoperability. This case was not about interoperability, as Google intended to design Android to be incompatible with Java. Moreover, Google's use of the same method and class names in Java API package files was not innovative since it merely involved verbatim copying. In short, this decision is a victory for innovators and original creators of content.

Brian Dunne, Olavi Dunne LLP

In a doctrinal vacuum, there is nothing legally sensible about holding that the structure and organization of API code — or software code generally, for that matter — is copyrightable expression. Software code is essentially functional. API code, a subset of software code, is likewise essentially functional. In a very real sense, Oracle v. Google and the software copyright precedents it relies upon are legally indefensible.

As a policy matter, however, the Oracle v. Google decision is hardly innovation kryptonite. Far from it. Against an IP backdrop where patent protection for functional software code is uncertain at best, it's critically important that valuable software innovations receive some form of legal protection. There are certainly square peg/round hole problems with holding that copyright, rather than patent, should supply this protection, but as a policy matter software copyright is immensely preferable to the most likely third choice for protecting software innovation — trade secrecy. Society benefits from innovators channeling their time and effort into code quality, not code obfuscation.

As far as interoperability, there's no question that API copyrightability creates an obstacle. APIs are essentially functional, and their function is to enable interoperability. But holding that APIs are copyrightable doesn't bar interoperability — it simply creates an interoperability tax. In many cases, this interoperability tax is paid to a real innovator, and its magnitude is correlated with innovation value. There are certainly exceptions to the foregoing, but they are better dealt with through fair use than through copyrightability.

Mark Duell, Honda Patents & Technologies North America LLC

The court decision in Google v. Oracle certainly will have a short-term impact on the interoperability of devices using Android in a Java environment. However, I think the idea that the decision will stifle innovation is incorrect. If anything, the decision will force Google to design around the restrictions on its use of Oracle's APIs. It is often this type of situation that forces an individual or a company to create a solution to an external problem, and I would expect Google to be innovative enough to engineer a solution that may actually improve the technology moving forward.

The bigger question is whether the court was correct in finding copyright protected Oracle's APIs. On this front, it is certainly questionable whether there was enough creative expression in these APIs to support copyright. This is the type of innovation that should be protectable by a patent with its more restrictive infringement rules and shorter term. Unfortunately, in the current environment where courts are running amok applying 101/Alice to just about everything, copyright certainly proved to be a useful backstop for Oracle.

Aaron Cooper, Covington & Burling LLP

The debate over the scope of patent-eligible subject matter has been the hot IP topic at the Supreme Court and the Federal Circuit in recent years. The court has granted certiorari and issued opinions in four Section 101 cases since 2010 — Bilski, Mayo, Myriad and Alice. Here, the court had the opportunity to opine on a parallel issue concerning copyright eligibility, but exercised restraint. Given the criticism leveled at the court for its patent subject matter eligibility opinions, it is somewhat surprising to hear concerns now about the court's decision not to wade into the contours of copyright protection. Unlike with questions of patent law, copyright issues can be considered by the regional circuits. Indeed, the Federal Circuit applied Ninth Circuit law in this case. That is one of numerous reasons for the court to deny the petition — another being the solicitor general's recommendation it be denied — and it should also serve to minimize any concerns.

Padmaja Chinta, Cittone & Chinta LLP

Unlike patents, the eligibility threshold for copyrights is pretty low. If your drawing of a stick figure is creative enough for a copyright, APIs certainly cross this existing threshold. Any concern that keeping and interpreting the eligibility threshold as-is will somehow stifle innovation in the software sector is misplaced. There is no functionality exception for copyrights like there is for trade dress. Indeed, all software is functional by nature. But software copyrights have been around for a while. And APIs have also been around for a while. Developers have been creating, changing and using APIs for a while. That has not stopped software innovation. This is not an instance of someone monopolizing the entire Java programming language or the mechanical specifications of a computer or the human DNA. While APIs are essential, they can easily be built in more than one way. And Java is available as an open source license. If anything, coming up with a different way of operation or method-call for the API is creative and will foster innovation if protected. That said, the debate will rage on for some time.

Andrew W. Carter, Ocean Tomo LLC

I believe the real impact of this denial cannot be judged until a decision on fair use is reached. And I don't believe there is any consensus on which way that decision will fall. After that, it would seem this case is likely to head back to the appellate level, and then potentially another petition for writ of certiorari will be filed. So the true impact is a long way from being decided.

Apart from the issues specific to this Google v. Oracle matter, this denial (along with others) suggests that the court's recent close involvement in IP may be on the decline. Based on my two-plus decades in this field, the last few years have been somewhat remarkable in the Supreme Court's involvement. Will the court now revert to a more hands-off policy in the IP space? Or did the court, given its heavy docket of key rulings (Obamacare, same-sex marriage, voter redistricting, clean air, and lethal injection) just decide to wait for a more interesting and/or well developed question to resolve? The next year will give us more insight into the Supreme Court's thoughts in this area.

Richard Baker, New England Intellectual Property LLC

In the realm of software patent law, there truly is nothing new under the sun, as we have returned right back to where we started 34 years ago, before Diamond v. Diehr. Software, along with its interfaces, is firmly covered by copyright law, as confirmed by Google v Oracle, and the patent coverage for software is allowable, after Alice, if one is careful in the claim drafting. And the fringes of the software community continue their ceaseless complaints about having to pay inventors and engineers for their work.

Intellectual property protection for software has provided an environment that has encouraged innovation and generated the $388 billion per year software industry in only 70 years. Companies are willing to pay the salaries of software engineers, knowing that there are protections in place to prevent the unlicensed copying of software. And venture capitalists and other investors are willing to invest in software companies knowing that the products will not be copied without the threat of legal enforcement of the intellectual property laws.

Interoperability is available to companies based on licenses or strategic decisions from the software owners to open their interfaces. This system has worked well, and Google v. Oracle simply retains a successful IP protection paradigm already in place.

****

The opinions expressed are those of the author(s) and do not necessarily reflect the views of the firms, their clients, or Portfolio Media Inc., or any of their respective affiliates. This article is for general information purposes and is not intended to be and should not be taken as legal advice.