No. 18-956

IN THE

Supreme Court of the United States

GOOGLE LLC,

Petitioner,

v.

ORACLE AMERICA, INC.,

Respondent.

On Petition for a Writ of Certiorari
to the United States Court of Appeals
for the Federal Circuit

BRIEF FOR THE AMERICAN ANTITRUST INSTITUTE AS AMICUS CURIAE IN SUPPORT OF PETITIONER

RANDY M. STUTZ
Counsel of Record
RICHARD M. BRUNELL
AMERICAN ANTITRUST INSTITUTE
1025 Connecticut Ave., NW
Suite 1000
Washington, DC 20036
(202) 905-5420
rstutz@antitrustinstitute.org

February 25, 2019

TABLE OF CONTENTS

Page

TABLE OF AUTHORITIES ii

INTEREST OF AMICUS CURIAE 1

SUMMARY OF ARGUMENT 2

ARGUMENT 3

I. THE FEDERAL CIRCUIT'S RULINGS ON COPYRIGHTABILITY AND FAIR USE INVOLVE QUESTIONS OF EXCEPTIONAL IMPORTANCE 3

A. Copyrights on Software Interfaces Risk Lock-in and Holdup 3

B. Affording Copyright Protection to Software Interfaces Will Cement Software-Based Monopolies 7

II. THE FEDERAL CIRCUIT'S RULINGS ON COPYRIGHTABILITY AND FAIR USE ARE INCONSISTENT WITH RULINGS BY THIS COURT AND OTHER COURTS OF APPEAL 10

A. The Federal Circuit Failed to Consider Interoperability and Compatibility Concerns 10

B. The Federal Circuit's Ruling on Transformativeness Guts the Fair-Use Doctrine as it Applies to Software 12

CONCLUSION 15

TABLE OF AUTHORITIES

Page

CASES

Apple, Inc. v. Motorola, Inc.,
757 F.3d 1286 (Fed. Cir. 2014) 5

Authors Guild v. Google, Inc.,
804 F.3d 202 (2d Cir. 2015) 13

Campbell v. Acuff-Rose Music, Inc.,
510 U.S. 569 (1994) 13

Commonwealth Sci. & Indus. Research Org. v. Cisco Sys., Inc., 809 F.3d 1295 (Fed. Cir. 2015) 5

eBay Inc. v. MercExchange, L.L.C.,
547 U.S. 388 (2006) 5

Harper & Row Publishers, Inc. v. Nation Enters.,
471 U.S. 539 (1985) 15

Lotus Dev. Corp. v. Borland Int'l, Inc.,
49 F.3d 807 (1st Cir. 1995) ..... *passim*

Oracle Am., Inc. v. Google Inc. (*Oracle I*),
750 F.3d 1339 (Fed. Cir. 2014) 2, 10

Oracle Am., Inc. v. Google LLC (*Oracle II*),
886 F.3d 1179 (Fed. Cir. 2018) 2, 11, 13

Oracle Am., Inc. v. Google Inc.,
2016 WL 3181206 (N.D. Cal. June 8, 2016) 13

Sega Enters. Ltd. v. Accolade, Inc.,
977 F.2d 1510 (9th Cir. 1992) 2, 3, 4, 10

Seltzer v. Green Day, Inc.,
725 F.3d 1170 (9th Cir. 2013) 13

Sony Computer Entm't, Inc. v. Connectix Corp.,
203 F.3d 596 (9th Cir. 2000).....14

Twentieth Century Music Corp. v. Aiken,
422 U.S. 151 (1975).....2

Universal City Studios, Inc. v. Corley,
273 F.3d 429 (2d Cir. 2001) 14

OTHER AUTHORITIES

Clark D. Asay, *Software's Copyright Anticommons*,
66 Emory L.J. 265 (2017) 6, 15

Brief for the United States as Amicus Curiae,
Google Inc. v. Oracle America, Inc.,
135 S. Ct. 2887 (2015) (No. 14-410).....10, 11

Brief of Amici Curiae the Computer &
Communications Industry Association and the
American Antitrust Institute, *Cisco Sys., Inc. v.*
Arista Networks, Inc., No. 2017-2145
(Fed Cir. filed Dec. 28, 2017).....9

BSA | The Software Alliance, *The $1 Trillion*
Economic Impact of Software (2016), *available at*
https://softwareimpact.bsa.org/pdf/Economic
_Impact_of_Software_Report.pdf.....2

Donald S. Chisum et al., *Last Frontier Conference*
Report on Copyright Protection of Computer
Software, 30 Jurimetrics 15 (1989) 3

Fed. Trade Comm'n, *The Evolving IP Marketplace:*
Aligning Patent Notice and Remedies with
Competition (2011), *available at*
https://www.ftc.gov/sites/default/files/documents/re
ports/evolving-ip-marketplace-aligning-patent-
notice-and-remedies-competition-report-federal-
trade/110307patentreport.pdf.....5

Fed. Trade Comm'n, *To Promote Innovation: The Proper Balance of Competition and Patent Law and Policy* (2003), *available at* https://www.ftc.gov/sites/default/files/documents/reports/promote-innovation-proper-balance-competition-and-patent-law-and-policy/innovationrpt.pdf.....6

Final Report of the Nat'l Commission on New Technological Uses of Copyrighted Works (1978) 3

Peter S. Menell, *Rise of the API Copyright Dead?: An Updated Epitaph for Copyright Protection of Network and Functional Features of Computer Software*, 31 Harv. J. L. & Tech. 305 (2018).....8, 9

William F. Patry, *Patry on Fair Use* (May 2018 Update).....14

Pamela Samuelson, *The Uneasy Case for Software Copyrights Revisited*, 79 Geo. Wash. L. Rev. 1746 (2011).....4

Carl Shapiro & Hal R. Varian, *Information Rules* (1999).....7, 8, 9

U.S. Copyright Office, *Software-Enabled Consumer Products* (December 2016), *available at* https://www.copyright.gov/policy/software/software-full-report.pdf 2, 11, 12

INTEREST OF AMICUS CURIAE

The American Antitrust Institute ("AAI") is an independent non-profit organization devoted to promoting competition that protects consumers, businesses, and society.[1] See http://www.antitrustinstitute.org. AAI serves the public through research, education, and advocacy on the benefits of competition and the use of antitrust enforcement as a vital component of national and international competition policy. AAI also seeks to ensure that intellectual property laws are interpreted and applied in a manner that reflects their ultimate goals of promoting innovation, competition, and consumer welfare.

AAI submits this brief because the Federal Circuit's application of the copyright laws to computer software interfaces undermines those goals and threatens substantial competitive harm in software-dependent markets throughout the U.S. economy. The public is harmed when even a large company like Google must pay royalties to license software interfaces. But this case also has implications for whether start-up firms that may challenge entrenched incumbents (like Google itself) will be deterred from doing so because of the barrier to entry created by the Federal Circuit's overprotection of software interfaces.

---

[1] Counsel of record have received timely notice of intent to file this brief under Rule 37.2(a) and all parties gave written consent to its filing. No counsel for a party has authored this brief in whole or in part, and no person other than amicus curiae has made a monetary contribution to fund its preparation or submission. Individual views of members of AAI's Board of Directors or Advisory Board may differ from AAI's positions.

SUMMARY OF ARGUMENT

This Court should grant certiorari to overturn the Federal Circuit's rulings that software interfaces like the Java applications programming interface (API) declarations are entitled to copyright protection and that the fair-use defense cannot apply to software innovation built on a copyrighted interface unless the innovation changes the meaning or expression of the copied elements. *Oracle Am., Inc. v. Google Inc.* (*Oracle I*), 750 F.3d 1339 (Fed. Cir. 2014); *Oracle Am., Inc. v. Google LLC* (*Oracle II*), 886 F.3d 1179 (Fed. Cir. 2018).

1. These rulings present questions of exceptional importance because, if not overturned, they may slow innovation and competition in software-dependent markets, which are pervasive in the U.S. economy. *See, e.g.*, BSA | The Software Alliance, *The $1 Trillion Economic Impact of Software* 3–4 (2016); U.S. Copyright Office, *Software-Enabled Consumer Products* 3 (December 2016) (noting that "[s]oftware is now nearly ubiquitous," including in consumer products). Indeed, the rulings may cement software-based monopolies.

2. The Federal Circuit rulings are also inconsistent with rulings of this Court, which recognize that copyright law seeks to promote innovation and consumer welfare by preserving a balance between exclusive rights and competition. *E.g., Twentieth Century Music Corp. v. Aiken*, 422 U.S. 151, 156 (1975). And they are inconsistent with rulings of other courts of appeal that recognize the importance of compatibility and interoperability concerns in evaluating copyrightability, fair use, and the merger doctrince. *E.g., Sega Enters. Ltd. v. Accolade, Inc.*, 977 F.2d 1510, 1524 (9th Cir. 1992).

ARGUMENT

I. THE FEDERAL CIRCUIT'S RULINGS ON COPYRIGHTABILITY AND FAIR USE INVOLVE QUESTIONS OF EXCEPTIONAL IMPORTANCE

A. Copyrights on Software Interfaces Risk Lock-in and Holdup

Congress extended copyright to software in 1980 as a compromise among possible alternatives. Liberalizing patent protection, the availability of which was then unclear, would have gone too far. Defining a new, *sui generis* protection threatened to upset traditions of overarching patent and copyright laws. And affording no protection would have required the software industry to rely on contract, trade secret, or other state laws. *See Final Report of the Nat'l Commission on New Technological Uses of Copyrighted Works* 16–19 (1978).

In the first decade after Congress made its choice, a group of leading intellectual property scholars observed that "Congress . . . has left to the courts the difficult task[] of determining how to apply copyright to computer programs," and "[c]ourts have generally articulated traditional copyright standards for determining the scope of protection." Donald S. Chisum et al., *Last Frontier Conference Report on Copyright Protection of Computer Software*, 30 Jurimetrics 15, 16–17 (1989). But applying concepts designed for literary works to computer software can be like trying "to fit the proverbial square peg in a round hole." *Sega*, 977 F.2d at 1524 (internal quotation omitted); *see Lotus Dev. Corp. v. Borland Int'l, Inc.*, 49 F.3d 807, 820 (1st Cir. 1995) (Boudin, J., concurring) (likening difficulties of applying copyright law to computer programs to

"assembling a jigsaw puzzle whose pieces do not quite fit").

The problem is that "computer programs are, in essence, utilitarian articles—articles that accomplish tasks." *Sega*, 977 F.2d at 1524; *see Lotus*, 49 F.3d at 819 (Boudin, J., concurring) ("The computer program is a *means* for causing something to happen; it has a mechanical utility, an instrumental role, in accomplishing the world's work."). Thus, "[c]omputer programs pose unique problems for the application of the 'idea/expression distinction' that determines the extent of copyright protection." *Sega*, 977 F.2d at 1524. More generally, as Judge Boudin explained, "[u]tility does not bar copyright (dictionaries may be copyrighted), but it alters the calculus" for intellectual property protection. *Lotus*, 49 F.3d at 819. The benefit may be similar (stimulating the production of computer software),[2] "[b]ut the 'cost' side of the equation may be different [than for traditional literary works] where one places a very high value on public access to a useful innovation that may be the most efficient means of performing a given task." *Id.*

In particular, the calculus for protecting computer software "interfaces" like the Java API declarations at issue here or the command menu hierarchy at issue in *Lotus* is problematic at best. As Judge Boudin explained:

---

[2] *But see* Pamela Samuelson, *The Uneasy Case for Software Copyrights Revisited*, 79 Geo. Wash. L. Rev. 1746, 1776 (2011) (identifying "significant developments in the software industry [that] raise questions about how important copyright protection now is to enabling developers to recoup their R&D investments in software").

> Requests for the protection of computer menus present the concern with fencing off access to the commons in an acute form. A new menu may be a creative work, but over time its importance may come to reside more in the investment that has been made by *users* in learning the menu and in building their own mini-programs—macros—in reliance upon the menu. Better typewriter keyboard layouts may exist, but the familiar QWERTY keyboard dominates the market because that is what everyone has learned to use.

Id. at 819–20.

The problem of *patents* on software or other technologies that become elements of industry standards is well known. *See generally* Fed. Trade Comm'n, *The Evolving IP Marketplace: Aligning Patent Notice and Remedies with Competition* 234 (2011). Product manufacturers can become locked-in to the standard and thereby susceptible to patent "holdup," with the result that royalties are excessive and innovation by manufacturers is discouraged. *See id.* at 227. Courts have adjusted patent remedies to avoid such holdup. *See Commonwealth Sci. & Indus. Research Org. v. Cisco Sys., Inc.*, 809 F.3d 1295, 1305 (Fed. Cir. 2015); *Apple, Inc. v. Motorola, Inc.,* 757 F.3d 1286, 1331–32 (Fed. Cir. 2014), *overruled on other grounds by Williamson v. Citrix Online, LLC*, 792 F.3d 1339 (Fed. Cir. 2015) (en banc); *see also eBay Inc. v. MercExchange, L.L.C.*, 547 U.S. 388, 396 (2006) (Kennedy, J., concurring) (cautioning against injunctive relief "[w]hen the patented invention is but a small component of the product the companies seek to produce and the threat of an injunction is employed simply for undue leverage in negotiations").

A similar problem arises with copyrighted software interfaces. Copyright on largely functional elements of software that become an industry standard gives a copyright holder anticompetitive power to thwart or tax innovative developments that build upon the elements, and to misappropriate for itself investments by users or developers in learning those elements. *Cf. Lotus*, 49 F.3d at 821 ("[I]t is hard to see why customers who have learned the Lotus menu and devised macros for it should remain captives of Lotus because of an investment in learning made by the users and not by Lotus."). Even if the copyrighted elements are not as essential and the lock-in not as severe as with a standard-essential patent, the anticompetitive harm from a copyright holder's ability to raise the costs of the innovative developments—to the detriment of new entrants, customers of the incumbent, and the public at large—is similar and appropriately cabined by a liberal reading of § 102(b) or the fair-use defense.

Indeed, absent a robust "method of operation" or "merger" exception, or fair-use defense, the risk of copyright holdup seems likely to increase as software development becomes increasingly collaborative and "any given piece of software may include dozens, hundreds, or even thousands of copyright holders." Clark D. Asay, *Software's Copyright Anticommons*, 66 Emory L.J. 265, 279 (2017). The "building-block approach to software development . . . means that some copyright holder of a software object within a particular software stack could become an obstacle to the entire stack's use." *Id.* at 314; *cf.* Fed. Trade Comm'n, *To Promote Innovation: The Proper Balance of Competition and Patent Law and Policy*, ch. 3, at 44 (2003) ("innovation [in software] occurs cumulatively").

B. Affording Copyright Protection to Software Interfaces Will Cement Software-Based Monopolies

Copyright protection of software interfaces is particularly anticompetitive because it tends to prevent new entrants from challenging dominant incumbent platforms protected by network effects.

The multisided desktop and mobile operating system markets that serve as a backdrop to this case illustrate the stakes. According to "Metcalfe's Law," the proportional value to a network of a user's investment in joining the network is the *square* of the number of users who do so, such that "a tenfold increase in the size of the network leads to a hundredfold increase in its value." Carl Shapiro & Hal R. Varian, *Information Rules* 184 (1999). In the authors' example, if a network that has a $1 value to a single user increases to 10 users, then the *network*'s total value increases to *$100*. *Id*. In an operating system environment, both consumers and software developers (as well as hardware and other complementors) invest in learning system software, adding several different dimensions of value to the network.

Software-based markets are characterized by strong positive network effects, which means lock-in increases over time because switching costs increase as the network size increases and network participants make greater investments in training to learn the system. *See id*. at 121 ("[T]he training costs associated with replicating one's proficiency with a familiar piece of software tend to grow the more experience

one has with the familiar program.").[3] New entrants seeking to introduce a rival operating system must overcome the costs of inducing *both* consumers and software developers (as well as complementors) to switch to the new network. *Id*. at 184 ("The challenge to companies seeking to introduce new but incompatible technology into the market is to build network size by overcoming the *collective switching costs*—that is, the combined switching costs of all users."); Peter S. Menell, *Rise of the API Copyright Dead?: An Updated Epitaph for Copyright Protection of Network and Functional Features of Computer Software*, 31 Harv. J. L. & Tech. 305, 458 (2018) ("companies seeking to leapfrog a widely adopted standard face substantial risk" and must not only invent a better platform and devise a strategy to migrate consumers away from the dominant platform, but also "encourag[e] other software and complementary product developers to build for the new platform").

Accordingly, "[i]n many information industries, collective switching costs are the biggest single force working in favor of incumbents." Shapiro & Varian, *supra*, at 184; *see id*. at 185–86 (explaining that inefficient QWERTY keyboard layout persists because "the *human* component of the system" raises collective switching costs and creates significant difficulties for coordinating a move to superior technology). And "[w]orse yet for would-be entrants and innovators, switching costs work in a nonlinear way: convincing ten people connected in a network to switch to your

[3] Training costs are not the only switching costs created by network effects. Investments in the software itself (apart from training), hardware, durable complementary assets, and information systems also give rise to switching costs. Shapiro & Varian, *supra*, at 184.

incompatible network is more than ten times as hard as getting one customer to switch. But you need all ten, or most of them: no one will want to be the first to give up the [incumbent] network externalities and risk being stranded." *Id.* at 184–85.

In an important sense, then, this Court's decision whether to let stand the Federal Circuit's rulings will affect whether mature, software-driven markets with strong incumbents are contestable by entrepreneurs. "Leaving API design specifications outside of copyright protection enables entrepreneurs seeking to improve on successful platforms to build bridges for users and programmers," which "avoids excess inertia and accommodates creative destruction and evolution in those areas where the proprietor of the standard platform lacks patent protection." Menell, *supra*, at 468. However, if copyright owners can appropriate developers' training investments by asserting copyright protection over interfaces, then collective switching costs can make it virtually impossible for entrepreneurial rival networks to launch, grow, and eventually challenge established incumbents. *See* Shapiro & Varian, *supra*, at 184, 195 ("[S]ometimes this kind of barrier can be insurmountable. Incumbents with intellectual property rights over an older generation of technology may have the ability to unilaterally blockade a migration path . . . [and] stop rivals in their tracks[.]").

The paradigm is not limited to desktop and mobile operating systems, but rather applies wherever a dominant incumbent asserts copyright protection over the functional aspects of software interfaces. *See, e.g.*, Brief of Amici Curiae the Computer & Communications Industry Association and the American Antitrust Institute, *Cisco Sys., Inc. v. Arista Networks, Inc.*, No.

2017-2145 (Fed Cir. filed Dec. 28, 2017) (explaining why copyright protection of command line interface commands has existential consequences for competition against dominant firm in market for network switches).

II. THE FEDERAL CIRCUIT'S RULINGS ON COPYRIGHTABILITY AND FAIR USE ARE INCONSISTENT WITH RULINGS BY THIS COURT AND OTHER COURTS OF APPEAL

A. The Federal Circuit Failed to Consider Interoperability and Compatibility Concerns

In rejecting the district court's determination that the Java API declarations constituted an unprotectable "method of operation," the Federal Circuit dismissed Google's arguments about interoperability and compatibility as irrelevant to copyrightability. *See Oracle I,* 750 F.3d at 1368–72. This was error. It is inconsistent with the First Circuit's conclusion that the command menu hierarchy in *Lotus* was not copyrightable, notwithstanding expressive content. *See Lotus*, 49 F.3d at 817 ("That the Lotus command menu hierarchy is a 'method of operation' becomes clearer when one considers program compatibility."). And it conflicts with the Ninth Circuit's holding that "functional requirements for compatibility . . . are not protected by copyright" under § 102(b). *Sega*, 977 F.2d at 1522.

To be sure, the Federal Circuit said that concerns about compatibility and interoperability may be relevant to fair use. *See Oracle I*, 750 F.3d at 1372, 1377. So did the United States. *See* Brief for the United States as Amicus Curiae 17, *Google Inc. v. Oracle America, Inc.*, 135 S. Ct. 2887 (2015) (No. 14-410)

(hereinafter 14-410 U.S. Brief) (interoperability and lock-in concerns are "substantial and important" but "are far better addressed through the fair-use doctrine"); *see also Lotus*, 49 F.3d at 821 (Boudin, J., concurring) (suggesting that fair use was alternative, albeit inferior, doctrinal hook to ensure that users are not locked into de facto standards).

Yet in its fair-use decision, the Federal Circuit dismissed compatibility or interoperability considerations. The court framed Google's compatibility argument as, "Google sought 'to capitalize on the fact that software developers were already trained and experienced in using the Java API packages at issue.'" *Oracle II,* 886 F.3d at 1206 (quoting *Oracle I*, 750 F.3d at 1371). "But," the court said, "there is no inherent right to copy in order to capitalize on the popularity of the copyrighted work or to meet the expectations of intended customers." *Id.* at 1206–07.

The Federal Circuit misapprehended the compatibility point. It is not about free-riding, but whether the public is served insofar as copying the API declarations gives developers "an option to exploit their own prior investment in learning" the packages rather than remain captives of the copyright owner. *Lotus*, 49 F.3d at 821 (Boudin, J., concurring); *cf.* 14-410 U.S. Brief at 17 (noting petitioner's argument that copying "promoted innovation by enabling programmers to switch more easily to another platform"). And by the Federal Circuit's own prior reckoning in *Oracle I*, fostering interoperability of use should have been at least *relevant* to fair use.

The Federal Circuit's logic also defies the U.S Copyright Office's recent report on copyright issues related to software-enabled consumer products. U.S.

Copyright Office, *Software-Enabled Consumer Products, supra*. In the report, the Office "recognizes the importance of preserving the ability to develop products and services that can interoperate with software-enabled consumer products, and the goal of preserving competition in the marketplace." *Id*. at 52. Yet it concluded that legislation was not needed to achieve these goals because "faithful application of existing copyright law doctrines can preserve the twin principles of interoperability and competition." *Id*.

For example, the Office observed that because Section 102(b) exempts ideas or methods of operation *embodied or described* in computer code from copyright protection, "the Act does not prevent a competitor from studying code to determine the underlying methods it teaches, and from implementing those methods using *different* code than the original, to create an interoperable or competitive software-enabled consumer product." *Id*. at 53. And the Office explained that the doctrine of merger is "a promising avenue to permit copying for purposes of interoperability." *Id*. Moreover, "the Office believes that, in many cases, copying of appropriately limited amounts of code from one software-enabled product into a competitive one for purposes of compatibility and interoperability should also be found to be a fair use." *Id*. at 57; *see id*. at 59 ("proper application of [fair use] principles should ensure that copyright law preserves the ability to create interoperable products and services").

B. The Federal Circuit's Ruling on Transformativeness Guts the Fair-Use Doctrine as it Applies to Software

"[T]he goal of copyright, to promote science and the arts, is generally furthered by the creation of

transformative works." *Campbell v. Acuff-Rose Music, Inc.,* 510 U.S. 569, 579 (1994). But the Federal Circuit's fair-use ruling guts the ability of the fair-use doctrine to promote innovation and competition in software-dependent markets by rejecting interoperability concerns, as noted above, and by holding that, no matter how innovative the new software, it does not qualify as a transformative use if "there are no changes to the expressive content or message" of the elements that are copied. *Oracle II*, 886 F.3d at 1201–02. As the district court noted, "[i]f this were enough to defeat fair use, it would be impossible ever to duplicate declaring code as fair use." *Oracle Am., Inc. v. Google Inc.*, 2016 WL 3181206, at *8 (N.D. Cal. June 8, 2016).

The Federal Circuit relied on *Seltzer v. Green Day, Inc.*, 725 F.3d 1170 (9th Cir. 2013), for the proposition that "a work is not transformative where the user 'makes no alteration to the *expressive content or message* of the original work.'" *Oracle II*, 886 F.3d at 1201 (quoting *Seltzer*, 725 F.3d at 1177). But *Seltzer* actually said, "In the *typical* 'non-transformative' case, the use is one which makes no alteration to the expressive content or message of the original work." *Seltzer*, 725 F.3d at 1177 (emphasis added and omitted). *Seltzer* did not involve software code, nor did any of the other cases the Federal Circuit cited.

Works can also be transformative if they expand the utility of copyrighted works. *See, e.g., Authors Guild v. Google, Inc.*, 804 F.3d 202, 214 (2d Cir. 2015) ("transformative use is one that communicates something new and different from the original *or expands its utility*") (emphasis added). The Federal Circuit's failure to recognize this point in the context of

computer software is perverse.[4] While expressive components of software may be protected by copyright (subject to § 102(b)), software's benefit is primarily functional and utilitarian. And software interfaces become standards (and are copied) because of their functional, not expressive, value. Not recognizing utilitarian transformations would enable the holder of a software interface copyright with the barest degree of expressive creativity to monopolize (or tax) broad swaths of commerce that incorporate the interface and would thwart the most significant, pro-competitive uses of the fair-use doctrine in software-dependent industries. *Cf.* William F. Patry, *Patry on Fair Use* § 6:7 (May 2018 Update) (copyright only concerned with harm "caused by the use of expression"); U.S. Copyright Office, *supra*, at 57–58 (question is "whether the use is principally for the purpose of exploiting the creativity of the original author of the code"; "interoperability is a favored purpose") (internal quotation marks omitted).

Moreover, although the Federal Circuit claimed otherwise, reimplementing the declaring code itself changes the "message" of the code. *Cf. Universal City Studios, Inc. v. Corley*, 273 F.3d 429, 446–49 (2d Cir. 2001) (recognizing computer code itself as a form of speech); *Sony Computer Entm't, Inc. v. Connectix Corp.*, 203 F.3d 596, 606–07 (9th Cir. 2000)

---

[4] The Federal Circuit did acknowledge that placing a copyrighted work in a new context to serve a different purpose may be transformative. *Oracle II*, 886 F.3d at 1202. But it concluded that copying elements of a software program to develop a new operating system for a new category of products (smartphones) would not serve a different purpose. If not *ipse dixit*, this conclusion can only be explained by the court's giving dispositive weight to whether there is a change in message.

("Connectix's drafting of entirely new object code for its VGS program [is] transformative, despite the similarities in function and screen output."). As Professor Asay points out, "Software interfaces" like Java's API packages "are strictly functional in carrying out the specified functions and facilitating communication between software products. . . . Hence, whatever creativity interfaces entail only becomes present and relevant when they are paired with the software that implements them." Asay, *supra*, at 321.

The Federal Circuit's fair-use ruling prevents the fair-use doctrine from acting as a safety valve "to avoid rigid application of the copyright statute [in the software context] when . . . it would stifle the very creativity which that law is designed to foster." *Harper & Row Publishers, Inc. v. Nation Enters.*, 471 U.S. 539, 550 n.3 (1985) (internal quotation omitted). Accordingly, it not only warrants certiorari itself but also supports certiorari on the Federal Circuit's categorical ruling that API declarations with a minimum of expressive creativity are copyrightable in the first place.

CONCLUSION

For the foregoing reasons the Court should grant Google's Petition for Certiorari.

Respectfully submitted,

RANDY M. STUTZ
Counsel of Record
RICHARD M. BRUNELL
AMERICAN ANTITRUST INSTITUTE
1025 Connecticut Ave., NW
Suite 1000
Washington, DC 20036
(202) 905-5420
rstutz@antitrustinstitute.org