

No. 18-956

IN THE
Supreme Court of the United States

GOOGLE LLC,

Petitioner,

v.

ORACLE AMERICA, INC.,

Respondent.

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

REPLY BRIEF FOR THE PETITIONER

Kent Walker
Catherine Lacavera
Renny Hwang
GOOGLE LLC
1600 Amphitheatre Parkway
Mountain View, CA 94043

Lisa S. Blatt
David M. Krinsky
Sarah M. Harris
Meng Jia Yang
WILLIAMS & CONNOLLY LLP
725 Twelfth Street, N.W.
Washington, DC 20005

Robert A. Van Nest
Christa M. Anderson
Eugene M. Paige
Reid P. Mullen
KEKER, VAN NEST
& PETERS LLP
633 Battery Street
San Francisco, CA 94111

Thomas C. Goldstein
Counsel of Record
Sarah E. Harrington
Kevin K. Russell
Daniel Woofter
Erica Oleszczuk Evans
GOLDSTEIN & RUSSELL, P.C.
7475 Wisconsin Avenue
Suite 850
Bethesda, MD 20814
(202) 362-0636
tg@goldsteinrussell.com

Michael S. Kwun
KWUN BHANSALI LAZARUS LLP
555 Montgomery Street
Suite 750
San Francisco, CA 94111

Bruce W. Baber
Marisa C. Maleck
KING & SPALDING LLP
1180 Peachtree Street, N.E.
Atlanta, GA 30309

TABLE OF CONTENTS

TABLE OF AUTHORITIES	iii
REPLY BRIEF FOR THE PETITIONER.....	1
I. The Copyright In Java SE Does Not Extend Protection To The Declarations	4
A. The Declarations Are A “Method Of Operation” Excluded By Section 102(b)	4
B. The Merger Doctrine Forecloses Copyright Protection For The Declarations.....	7
C. There Is No Merit To Oracle’s Reliance On Section 101.....	10
D. Oracle Errs In Arguing That Google Used Both Too Few And Too Many Of The Java SE Declarations	11
E. Copyright Protection Does Not Extend To The Java SE Filing System.....	13
II. There Is No Basis To Overturn The Jury’s Finding That Google’s Reuse Of The Declarations Was Fair Use	14
A. The Federal Circuit Applied The Wrong Standard Of Review	14
B. Substantial Evidence Supported The Jury’s Fair-Use Verdict.....	16
1. Oracle Fails to Grapple with Evidence of Industry Practice and Its Own Prior Positions Supporting the Reasonableness of Google’s Conduct	16

2. The Jury Could Reasonably Have Found That Google's Reuse Was Transformative	17
3. The Jury Was Entitled to Find No Market Harm to Java SE	21
CONCLUSION	23

TABLE OF AUTHORITIES

Cases

<i>Baker v. Selden</i> , 101 U.S. 99 (1880)	8, 9, 11
<i>Bateman v. Mnemonics, Inc.</i> , 79 F.3d 1532 (11th Cir. 1996)	2
<i>Campbell v. Acuff-Rose Music, Inc.</i> , 510 U.S. 569 (1994)	<i>passim</i>
<i>Computer Assocs. Int’l, Inc. v. Altai, Inc.</i> , 982 F.2d 693 (2d Cir. 1992)	5
<i>Direct Mktg. Ass’n v. Brohl</i> , 575 U.S. 1 (2015)	5
<i>Folsom v. Marsh</i> , 9 F. Cas. 342 (C.C.D. Mass. 1841)	15
<i>Golan v. Holder</i> , 565 U.S. 302 (2012)	5
<i>Harper & Row, Publishers, Inc. v. Nation Enters.</i> , 471 U.S. 539 (1985)	15, 16, 23
<i>Lexmark Int’l, Inc. v. Static Control Components, Inc.</i> , 387 F.3d 522 (6th Cir. 2004)	2
<i>Lotus Dev. Corp. v. Borland Int’l, Inc.</i> , 49 F.3d 807 (1st Cir. 1995)	2, 6, 10
<i>Reeves v. Sanderson Plumbing Prods., Inc.</i> , 530 U.S. 133 (2000)	16
<i>Sega Enters. Ltd. v. Accolade, Inc.</i> , 977 F.2d 1510 (9th Cir. 1992)	3, 11
<i>Sony Computer Entm’t, Inc. v. Connectix Corp.</i> , 203 F.3d 596 (9th Cir. 2000)	3, 22

<i>Sony Corp. of Am. v. Universal City Studios, Inc.</i> , 464 U.S. 417 (1984)	21
<i>Southco, Inc. v. Kanebridge Corp.</i> , 390 F.3d 276 (3d Cir. 2004)	10
<i>Tennant v. Peoria & Pekin Union Ry.</i> , 321 U.S. 29 (1944)	16
<i>Twentieth Century Music Corp. v. Aiken</i> , 422 U.S. 151 (1975)	21
<i>Whelan Assocs. v. Jaslow Dental Lab., Inc.</i> , 797 F.2d 1222 (3d Cir. 1986)	5

Statutes

17 U.S.C. § 101	6, 10
17 U.S.C. § 102(b)	<i>passim</i>
17 U.S.C. § 302(a)	8

Rules

Fed. R. Civ. P. 50(a)(1)	15
--------------------------------	----

Other Authorities

H.R. Rep. No. 94-1476 (1976)	5
Pierre N. Leval, <i>Toward a Fair Use Standard</i> , 103 Harv. L. Rev. 1105 (1990)	18
S. Rep. No. 94-473 (1975)	5, 23
U.S. Copyright Office, <i>Software-Enabled Consumer Products</i> (Dec. 2016)	22

REPLY BRIEF FOR THE PETITIONER

Software interfaces convey information and instructions between computer programs and platforms. They are the digital equivalent of an electrical outlet, a car's steering wheel, and a QWERTY keyboard. Imagine if every piece of equipment had its own unique plug, every car a different way to steer, and every computer its own distinct way to type.

The same challenges would arise in the digital world if every computer program had its own proprietary way to share information. But the opposite has been true—and the software industry has thrived—because of the “long-standing, ubiquitous practice” of reusing software interfaces. Computer Scientists Br. 3. In this interoperable era, countless applications and platforms reuse existing interfaces to share information. Also, innumerable new applications “reimplement” earlier software by reusing the original interfaces that allow information to be sent and received. See Software & System Developers Br. 14-16; IBM Br. 16; Microsoft Br. 14; Computer Scientists Br. 17-22. Java SE *itself* reuses interfaces used in earlier languages, including those corresponding to declarations for which it now asserts copyright infringement *in this case*. See JA154-157; JA179; JA211; Computer Scientists Br. 20; see also Rimini St. Br. 6-13; R St. Br. 17.

In Android, Google reimplemented certain “methods”—*i.e.*, shortcut programs—of Java SE. Google created Android's implementing code, adapting and improving the methods for use in an entirely new category of devices: smartphones. Google reused certain “declarations” from Java SE. Google's reason for reusing those declarations had nothing to do with

desperation, insufficient resources, or a lack of skill. Google simply had no other choice: only those declarations would respond correctly to the calls written by third-party developers. Pet. App. 221a, 264a; *see* Google Br. 3-6 & n.4, 8-10.

Oracle does not dispute that Google reused the minimum amount of material required to permit developers to use those calls. Google reused only declarations (not implementing code), and even then only those “required to meet [the] expectations of Java programmers” creating smartphone applications. JA69. Google even changed the limited elements of the declarations that were not strictly dictated by the Java language. Google Br. 32-33 (discussing names used for input variables). Google also did not duplicate how the methods are ordered within the electronic files or the logical relationships between the methods. *Ibid.* (discussing how Android did not copy the manner in which Java SE methods incorporate each other).

The district court correctly rejected Oracle’s copyright infringement claim, on two grounds. First, it held that declarations are excluded from copyright protection by Section 102(b) of the Copyright Act, 17 U.S.C. § 102(b), as either a “method of operation” or (under the merger doctrine) as one of only a few ways to express a method of operation. *Accord Lexmark Int’l, Inc. v. Static Control Components, Inc.*, 387 F.3d 522, 536 (6th Cir. 2004) (Sutton, J.) (copyright protection does not apply when “compatibility requires” literal copying); *Lotus Dev. Corp. v. Borland Int’l, Inc.*, 49 F.3d 807 (1st Cir. 1995), *aff’d by an equally divided Court*, 516 U.S. 233 (1996) (per curiam); *id.* at 821 (Boudin, J., concurring); *see also Bateman v. Mnemonics, Inc.*, 79 F.3d 1532, 1547 (11th Cir. 1996) (no protection

where copying was “dictated by compatibility requirements”).

Second, the district court found that there were “myriad” ways that the jury—which heard conflicting evidence on all the statutory fair-use factors and several other relevant considerations—could have deemed Google’s reuse of the declarations to be “fair use.” Pet. App. 117a; *accord Sony Computer Entm’t, Inc. v. Connectix Corp.*, 203 F.3d 596 (9th Cir. 2000); *Sega Enters. Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992).

Oracle’s contrary claim that Google’s limited reuse of the mandatory declarations is nonetheless copyright infringement amounts to an anti-competitive “hold up.” It seeks to lock developers into using the Java SE platform to write programs in the Java language. More broadly, if Oracle’s theory is adopted, the existing users of legacy software will often find it too difficult and costly to learn and implement a new set of instructions to transition to a better alternative. AAI Br. 7.

As Google’s amici stressed in urging this Court to grant certiorari, the long-settled practice of reusing interfaces means that a ruling in Oracle’s favor threatens to impose debilitating retroactive liability. Google Cert. Reply Br. 6-8 (collecting citations). It would inflict the most harm on small software developers—such as new entrants attempting to improve on existing software, who need to attract the users of earlier programs. *See, e.g.*, Software Innovators Cert. Br. 3; AAI Cert. Br. 9.

In this Court, the Nation’s leading computer scientists, technology companies, and copyright scholars

have come together in near consensus, explaining that overturning the district court’s rulings would seriously impede the development of platforms and computer software, and would be fundamentally inconsistent with basic copyright principles. For the reasons given by the amici and outlined below, this Court should reject Oracle’s arguments for deeming the declarations protected by copyright and for overturning the jury’s holistic determination that Google’s conduct was fair use.

I. The Copyright In Java SE Does Not Extend Protection To The Declarations.

A. The Declarations Are A “Method Of Operation” Excluded By Section 102(b).

Oracle argues that declarations (and interfaces more generally) do not describe or embody an “idea” or “method of operation” that is excluded from copyright protection either (1) directly by Section 102(b), or (2) under the merger doctrine as one of only a few available means of expressing an idea or method of operation. According to Oracle, all the listed phrases in Section 102(b) are merely synonyms for “idea.” Oracle Br. 24. Oracle then asserts that Java SE reflects only a single, overarching “idea”: “to provide a collection of modular programs that are helpfully organized and described to enable developers to use them in writing their apps.” *Id.* at 25.

In fact, Section 102(b) excludes from copyright protection not merely a program’s single overarching “idea” but also, *inter alia*, “any . . . procedure, process, system, [or] method of operation . . . described . . . or embodied” in a computer program. 17 U.S.C. § 102(b) (emphases added). There is a strong presumption that

different terms in a list have distinct meanings. *Direct Mktg. Ass'n v. Brohl*, 575 U.S. 1, 13 (2015). That presumption is sound here. In ordinary usage, an “idea” is a concept, which contrasts directly with, for example, a concrete “process.”

Oracle’s reading also fails to fulfill the statute’s purpose to exclude from copyright protection anything beyond the author’s original, creative expression. *Golan v. Holder*, 565 U.S. 302, 328 (2012). Congress enacted Section 102(b) to preclude “protection to the methodology *or processes* adopted by the programmer, rather than merely to the ‘writing’ expressing his ideas.” H.R. Rep. No. 94-1476, at 57 (1976) (emphasis added); S. Rep. No. 94-473, at 54 (1975). Java SE, for example, includes some 10,000 different short-cut programs. According to Oracle, it may claim an exclusive copyright to the “idea” underlying every one of those methods, as well as all of their individual processes, merely by combining them into a single computer program. That cannot be right.

Oracle’s view that Section 102(b) applies at an exceedingly high level of generality was adopted by one early decision, but the courts of appeals have uniformly rejected it since. A principal reason is that it fails to recognize that one program may reflect many “ideas” or “methods of operation.” *See, e.g., Computer Assocs. Int’l, Inc. v. Altai, Inc.*, 982 F.2d 693, 705 (2d Cir. 1992) (discussing *Whelan Assocs. v. Jaslow Dental Lab., Inc.*, 797 F.2d 1222, 1236 (3d Cir. 1986), as “widely criticized for being conceptually overbroad”); *see also* CCIA Br. 10-14; Menell Br. 28-29 & n.6.

Application of Section 102(b) is thus straightforward: Is the interface provided by the Java SE declarations a “method of operation” (or a “procedure,

process, [or] system”)? As noted, Oracle does not define “method of operation” or any of the other statutory terms, because it erroneously assumes they are synonyms. But it offers no reason to depart from the commonsense understanding that a method of operation “refers to the means by which a person operates something, whether it be a car, a food processor, or a computer.” *Lotus*, 49 F.3d at 815.

The declarations easily fit that understanding. They are a method of operation because they are for the *developers* to use, while the implementing code instructs the computer. *Compare* 17 U.S.C. § 102(b), *with id.* § 101; *see also* AIPLA Br. 13; Computer Scientists Br. 5-6; EFF Br. 26 (discussing Copyright Office’s policy of not allowing registration of computer “interfaces”). As the parties jointly stipulated after the Federal Circuit held that the declarations were protected by copyright, declarations “allow[] programmers to understand and make use of the prewritten programs,” whereas “implementing code provides step-by-step instructions that tell the computer how to perform the function specified by the declar[at]ions.” JA106-107. *Contra* Oracle Br. 27 (asserting that Google’s reading would exclude all computer code from copyright protection).

Oracle argues to the contrary that Section 102(b) embodies the “idea/expression” dichotomy, such that it excludes from copyright protection only the functionality of the “method of operation” rather than the actual computer instructions. Oracle Br. 26. An author thus cannot claim an exclusive right to a method of operation described in a work—such as a method for assembling a bicycle.

In fact, Section 102(b) goes beyond merely reflecting the idea/expression dichotomy. It affirmatively excludes certain computer code by its terms. Section 102(b) expressly excludes “any” “method of operation” that is “*embodied in*” (not merely *described* in) any computer program. 17 U.S.C. § 102(b) (emphasis added). The word “embodied” is naturally understood to refer to the computer instructions set forth in the program that make up a method of operation, which necessarily are themselves subject to exclusion from copyright protection under the statute.

B. The Merger Doctrine Forecloses Copyright Protection For The Declarations.

In any event, the declarations are excluded from copyright protection by the merger doctrine, which is itself an application of the idea/expression dichotomy. Merger provides that copyright protection does not apply when there are only a few available means of expression. Otherwise, the copyright in the expression would indirectly grant an exclusive right to the underlying functionality. *See* Google Br. 31.

The dispositive, undisputed fact in this case is thus that—as the district court found—the declarations cannot be written in any other way and still properly respond to the calls used by Java programmers. Pet. App. 221a, 264a; Computer Scientists Br. 7. Put another way, only those declarations provide the method of operation. And Oracle is inescapably claiming on the basis of its copyright the exclusive right to that method of operation. Section 102(b) precludes extending copyright protection that far.

Take Oracle's example of the *verify* method, which it no doubt selected because it contains an unusually complicated declaration. In *English*, Oracle describes the declaration in six different sentences that it could have written in numerous different ways. Oracle Br. 6. But much more important, in the *Java* language that *programmers* use, only the precise declaration that Sun used in Java SE will work. Both the English description and the Java declaration are expression; neither can confer on Oracle the exclusive right to the *verify* declaration's functionality.

Oracle's contrary approach cannot be reconciled with *Baker v. Selden*, 101 U.S. 99 (1880). Selden claimed copyright over not only an essay describing his accounting method, but also forms that were required to track income and expenses using that method. On Oracle's view, that claim should have prevailed, because the author's "idea" was merely to keep an accounting ledger, and the author's particular forms were not required to effectuate that general idea. But *Baker* rejected that claim. It reasoned that copyright protected the essay, but the author could not claim an exclusive right to forms necessary to use the system of accounting.

Oracle argues that Google's application of the merger doctrine relies on the popularity of the Java SE declarations, in violation of the principle that "[c]opyright in a work . . . subsists from its creation." 17 U.S.C. § 302(a). But everyone agrees that Oracle holds a valid copyright in the overall *work*, which is Java SE. The question is whether that copyright protection extends to an element of the work: the declarations.

Nothing in the statute provides that Section 102(b) and merger cannot be determined from the

perspective of the alleged infringer. Thus, in *Baker*, the Court considered whether a later user would need to use a form like Selden’s to perform the author’s method of accounting. Similarly, nothing in law or logic prevents a court from later determining that certain computer code is the only means of expressing an important function. That is in fact precisely how the authoritative CONTU Report anticipates merger being applied to computer software. Google Br. 30.

In any event, Google’s argument does not rely on events that occurred subsequent to the release of the Java SE libraries, and it does not depend on the declarations’ subsequent “popularity.” It is undisputed that once Sun made its conceptual choices about how the declarations should operate, *Sun itself* could express the declarations in only one way. The Java programming language permitted no other options. Thus, while Oracle stresses that the *conceptual decisions* behind the organization of the Java SE libraries are creative, it ignores the district court’s conclusion and the witness testimony that the *declarations themselves* unquestionably were not. *E.g.*, Pet. App. 221a, 264a; JA210-212.

The fact that Sun chose the declarations’ names does not dictate a different result. Names are not copyrightable. *See* Google Br. 29. Further, Sun “*wouldn’t want* names to be creative because as software developers, we’d expect to have the names in our programming libraries be descriptive and functional of their purpose.” JA211 (emphasis added); Computer Scientists Br. 8. Look no further than the evocatively entitled *max* and *verify* declarations.

In *Baker v. Selden*, this Court held that the author’s forms were not protected, notwithstanding that

the defendant could have used different names as column headers. 72 IP Scholars Br. 25. Here too, once Sun selected the names, those names could be expressed in the declarations in only one way. The Java programming language did not, for example, permit Sun in a single declaration to write the names of the relevant package, class, and method together, nor could Sun use a shorthand reference for the name.

C. There Is No Merit To Oracle’s Reliance On Section 101.

Oracle cites only one supposed statutory basis for its position. It argues that because Congress implicitly treated “computer program[s]” as “[l]iterary works,” 17 U.S.C. § 101, *ipso facto* this Court must treat every instruction in every program as if it were the narrative of a novel, Oracle Br. 21, 44. That is no answer, for at least two reasons: Declarations (as opposed to implementing code) cannot be written creatively, unlike a narrative, and Section 102(b) is an express limitation on the scope of the protection otherwise afforded to copyrighted works.

More broadly, Oracle’s analogy to literature also lacks merit. Not only do different kinds of literary works receive varying copyright protection, *Campbell v. Acuff-Rose Music, Inc.*, 510 U.S. 569, 586 (1994); see *Southco, Inc. v. Kanebridge Corp.*, 390 F.3d 276, 284 (3d Cir. 2004) (en banc) (part numbers dictated by a system—which are closely analogous to the rote declarations dictated by the Java language—receive *no* copyright protection because they are “purely functional”), but computer programs specifically aren’t analogous to literature, because they are highly functional, *e.g.*, *Lotus*, 49 F.3d at 819 (Boudin, J.,

concurring); *Sega Enters. Ltd. v. Accolade, Inc.*, 977 F.2d 1510, 1524 (9th Cir. 1992) (“[C]omputer programs are, in essence, utilitarian articles—articles that accomplish tasks.”).

Thus, great care must be taken not to permit the author of software to monopolize its functioning, a right that can only be granted through patent law, subject to the limits of novelty and non-obviousness. *Baker*, 101 U.S. at 102. Oracle’s position would circumvent patent law’s strict limits on exclusive software rights. Engine Br. 18-26. Only a low level of creativity is required to make a work technically “copyright eligible.” Copyrights also last decades longer than patents. The declarations of the Java SE libraries would be both obvious (given the long history of software interfaces) and patent-ineligible subject matter (because they simply establish a means to control an application).

D. Oracle Errs In Arguing That Google Used Both Too Few And Too Many Of The Java SE Declarations.

Oracle argues that if Google had reused only the 170 lines of code that Oracle admitted were a fair use, third-party developers still could have written *something* in the Java language—*i.e.*, an incredibly simplistic Java program. Oracle Br. 32; *see* 2016 Trial Transcript (Tr.) 1442-1446 (stipulating that a *minimum* of 170 would be required); *id.* at 2209-2210 (jury instructions). That is like saying that, with only a noun and verb, one could write an English sentence. It would nonetheless choke expression to remove from the English language several thousand words and phrases that are the only way to explain important concepts.

Oracle's approach would similarly prevent third-party developers from using the calls on which they depend to write applications. *See* JA67-68 (forbidding use of just *four* calls would require a Java developer to re-write 1,000 lines of computer code).

Oracle's further argument that Apple and Microsoft developed mobile operating systems without reusing any of the Java SE declarations, Oracle Br. 31, is misleading. For their own reasons, neither company chose the "free and open" Java programming language. So the Java SE declarations were not relevant. But both *did* reimplement *other* software. Microsoft Br. 14-15; Computer Scientists Br. 18-19.

Oracle next takes the opposite position that Google used *too few* declarations, because copyright supposedly required Google to reimplement all of the Java SE methods. It reasons that not every Java SE program will work on the Android platform, and not every Android program will work with Java SE. But *no* program uses even a small proportion of all of the methods in the Java SE or Android libraries.

To the extent Oracle is arguing that Android is not "interoperable," that is incorrect. Oracle assumes that interoperability requires that one program be compatible with its predecessor. "But complete compatibility is not necessary, or even desirable, to promote interoperability in software development." Computer Scientists Br. 13. It is common for a new and better application to replace an older one; the two do not "interoperate" at all in the way that Oracle uses that term. Here, the declarations that Google reused are interoperable with the corresponding calls used by developers to create smartphone applications.

Further, what Oracle demands would be functionally impossible, because Java SE and Android were created for very different devices. Google did not deliberately set out to make Android and Java SE “incompatible.” *Contra* Oracle Br. 51. Rather, Java SE applications were created for traditional desktop computers and servers with their monitors, keyboards, and mice; Android applications were created for smartphones with their touchscreens, cellular radios, GPS modules, and cameras. There is no copyright significance to the fact that Android does not include the useless feature of being interoperable with commands that those developers *would not* use. Moreover, it would have been functionally impossible to include all of the methods of Java SE on a smartphone, which is one of the reasons Oracle itself never succeeded in doing so. JA264-265.

E. Copyright Protection Does Not Extend To The Java SE Filing System.

Google did not infringe any separate, protected interest in the structure, sequence, and organization (SSO) of the Java SE libraries. Importantly, Oracle abandons the Federal Circuit’s theory that the Java SE libraries are a protected “compilation” of declaration names. Google Br. 33 & n.7. Here, Oracle claims at most that the declarations create a filing system for storing the methods on a computer in the package folders and class files. Google only incidentally and unavoidably duplicated that file structure when it reused the declarations. In turn, under the merger doctrine, copyright protection does not extend to that file structure because it is a “system” that can only be expressed through those exact declarations, 17 U.S.C. § 102(b); it *cannot* be embodied in any other form. Finding that Oracle had a protected interest in the structure would

prevent Google from reusing the declarations, precisely the result the merger doctrine forbids. Google Br. 31.

Oracle also does not dispute that no court has ever recognized a claim for something as purely functional as that filing system. Google Br. 32-33 & n.7. Courts have recognized claims relating to the logical relationship between the modules of a program. In the Java SE libraries, that relationship exists in the manner that the methods incorporate each other through implementing code. But Oracle does not contest that, because Google did not reuse the implementing code, it did not duplicate those relationships. *Ibid.*

Oracle also attempts to leave the false impression that the declarations are related to each other, such that they comprise a single segment of a program. In fact, they are scattered throughout the Java SE libraries. They are not protected by copyright any more than are historical facts scattered throughout an encyclopedia.

In any event, even if Oracle held a thin protected interest in either the declarations or the corresponding SSO, Google's reuse of the declarations was fair use.

II. There Is No Basis To Overturn The Jury's Finding That Google's Reuse Of The Declarations Was Fair Use.

A. The Federal Circuit Applied The Wrong Standard Of Review.

The ruling below is the first appellate decision *ever* to reverse a jury's fair-use verdict. It did so by erroneously applying a *de novo* standard of review to what it incorrectly viewed as the jury's "advisory only"

fair-use verdict. Pet. App. 24a. That error fatally infected the court's (and now Oracle's) evaluation of the evidence.

Because fair use is a fact-intensive inquiry, *Harper & Row, Publishers, Inc. v. Nation Enters.*, 471 U.S. 539, 560 (1985), the Federal Circuit itself ordered a second jury trial, Pet. App. 174a-184a. The properly instructed jury, JA279-290, heard two weeks of testimony and reviewed numerous documents, assessed all the evidence relevant to each statutory factor and the overarching purpose of copyright law, and returned a general verdict for Google.

The question is thus not whether the Federal Circuit thinks fair use was established, but whether the evidence was enough that a reasonable jury *could* have found fair use. *See* Fed. R. Civ. P. 50(a)(1); AIPLA Br. 25-28. It was. That is particularly true, because the jury was entitled to conduct its own weighing of the statutory factors and other non-statutory considerations it deemed to be relevant. *Campbell*, 510 U.S. at 578.

Oracle notes that in *Harper & Row* and *Folsom*, the reviewing courts determined fair use *de novo* as a matter of law. Oracle Br. 37. But neither case involved a jury verdict; a judge or chancery master made express factual findings, *Harper & Row*, 471 U.S. at 543; *Folsom v. Marsh*, 9 F. Cas. 342, 345 (C.C.D. Mass. 1841), and the reviewing court decided whether, as a matter of law, those facts established fair use. That also happens any time a court decides a claim on summary judgment. But when a jury renders a general verdict on *disputed* facts, a reviewing court must view the evidence in the light most favorable to the verdict, not perform its own *de novo* weighing of the evidence.

Relying on cases reviewing grants of summary judgment, the Federal Circuit assigned to itself the role of “reweigh[ing] on appeal the inferences to be drawn from th[e] record.” Pet. App. 18a (citation omitted). But a reviewing court is “not free to reweigh the evidence and set aside the jury verdict merely because the jury could have drawn different inferences or conclusions or because [the court] fe[lt] that other results are more reasonable.” *Tennant v. Peoria & Pekin Union Ry.*, 321 U.S. 29, 35 (1944); Cherry Professors Br. 13-14. Deciding which evidence in the two-week trial was persuasive and which witnesses were credible was the jury’s province, not the Federal Circuit’s. *Reeves v. Sanderson Plumbing Prods., Inc.*, 530 U.S. 133, 150-151 (2000).

B. Substantial Evidence Supported The Jury’s Fair-Use Verdict.*

1. *Oracle Fails to Grapple with Evidence of Industry Practice and Its Own Prior Positions Supporting the Reasonableness of Google’s Conduct.*

Fair-use determinations require an assessment not only of evidence about the four non-exclusive statutory factors, *Harper & Row*, 471 U.S. at 560, but also more broadly whether the use in question advances the goals of copyright, *Campbell*, 510 U.S. at 578. In undertaking its narrow role of determining whether there was sufficient evidence to support the verdict,

* Because Oracle and the United States focus almost exclusively on the first and fourth statutory factors, we do so as well in this reply. As explained in our opening brief, Google Br. 43-48, and above, the jury’s fair-use verdict is supported by the second and third factors as well.

the Federal Circuit failed to consider *any* of Google’s evidence on that point.

The jury heard that “many at Google (and Sun) understood that at least the declaring code and their [structure, sequence, and organization] were free to use and reimplement.” Pet. App. 106a. “Sun’s own CEO at the time, Jonathan Schwartz, testified . . . that a practice of duplicating declarations existed and that the competition was on” the implementing code, which Google did not copy. *Id.* at 106a-107a; JA55.

Oracle argues that Google should have licensed the declarations, a theory that would equally negate every defense to infringement. Oracle Br. 11. But there is no such thing as a “declaring code license.” That term was never uttered in either trial, appears nowhere in the record, and cannot even be found in the document to which Oracle refers. JA511. Oracle’s argument requires the Court to ignore abundant evidence, including testimony from former top executives responsible for the creation, development, licensing, and marketing of Java that the declarations were “[a]bsolutely” “marketed by Sun” “as free and open,” and that Sun “never” treated them as proprietary or “licensed [them] separately from the language.” JA124-125; JA129; *see* JA56; JA119-122; 2016 Tr. 332-335, 339, 493-494; Computer Scientists Br. 15. Indeed, Sun (and later Oracle) lauded Android despite knowing that Google had not taken a license. JA129; 2016 Tr. 556-558; Trial Exhibit (TX) 2352; TX2041 at 10-11.

2. *The Jury Could Reasonably Have Found That Google’s Reuse Was Transformative.*

Oracle does not dispute that the jury was correctly instructed on all aspects of fair use, including

transformativeness. JA281-290. Instead, it incorrectly asserts (without citation) that Google conceded the use of the declarations for the same purpose in a competing product. Oracle Br. 2.

a. The jury was entitled to find that Google’s reuse was transformative because Google “add[ed] something new” by “altering” the declarations “with new expression” in a new context. *Campbell*, 510 U.S. at 579. The *functional purpose* of the declarations was unchanged because the declarations by their nature can only ever connect specific calls with particular implementing code. All computer instructions can only ever perform one function. But the factor-one inquiry examines whether the “new work is ‘transformative,’” *ibid.*, and Android linked the declarations to entirely new implementing code to create a modern smartphone ecosystem unlike anything before it. *See id.* at 581-582 (discussing transformative effect of 2 Live Crew’s *added* expression). As Google’s technical expert testified, “the purpose” of the declarations was “different” in Android “because creating an application on the Android platform is a different context than creating an application on the laptop or desktop computer.” JA383-384. The jury was entitled to credit that testimony.

Google’s use of the declarations also fostered “the development of new ideas out of old,” Pierre N. Leval, *Toward a Fair Use Standard*, 103 Harv. L. Rev. 1105, 1109 (1990), by enabling developers trained in the Java programming language to create innovative programs for a modern smartphone for the first time, JA139-140; JA187; JA190-191; Pet. App. 196a; *infra* at 20-23. The Federal Circuit ignored that evidence.

b. Whether Android was a transformative product or a competitor to Java SE was hotly disputed. The jury heard evidence that Oracle “did not have a competitive product,” that “Android has not superseded Java SE,” and that Android and Java SE are “very different types of products” because Java SE is “just an applications programming framework” for desktops and servers, while Android is “an entire mobile operating stack that runs a smartphone.” JA676; JA255-257. That evidence was sufficient to find transformativeness, and that should be the end of it.

Oracle and the United States argue that Google’s reuse created a derivative work of Java SE by transferring declarations from big computers to smaller computers. That is wrong because it ignores fundamental differences in markets, technological architecture, and consumer usage of desktops and smartphones. The only derivative uses relevant to fair use are those the copyright owner could have created or licensed. *Campbell*, 510 U.S. at 592.

Oracle’s remaining arguments are nothing more than nitpicking the jury’s assessment of evidence about particular devices. Even on Oracle’s view of those contested facts, the jury was fully entitled to conclude that on net Android was transformative.

Oracle’s carefully worded contention that by 2005, “the Java *platform* was in over a billion mobile handsets,” Oracle Br. 11 (emphasis added), is extremely misleading. Those handsets used *Java ME*—Java Micro Edition—not Java SE. JA500 (cited at Oracle Br. 11) (noting one billion “wireless” units with “Java ME” installed); see JA239. Java ME is *not* the copyrighted work; it is a “different” work that predated the versions of Java SE at issue and did not include the same

set of declarations. 2016 Tr. 1479-1481, 1940-1941. Devices that used Java ME are *irrelevant* here.

Oracle contends it is “undisputed” that “Java SE was already being used in smartphones” before Android, pointing only to SavaJe and Danger. Oracle Br. 41-42 (citation omitted). But the jury heard testimony from former Sun executives that when Android was announced in 2007, *no* mobile phones were “running [Java] SE.” JA135-136; 2016 Tr. 580-581; *see* JA98. SavaJe had proven unsuccessful *before* the first Android phones were released. *See* JA235; JA650; JA671; *see also* JA171. The jury saw SavaJe’s phone and heard that it was not a true smartphone because it lacked a touchscreen and a QWERTY keyboard. *See* JA221; JA668-669; JA671-673.

The story is similar with Danger’s Sidekick, which also lacked a touchscreen. JA359-360; JA669; *see* 2016 Tr. 1074. Moreover, the jury heard that Danger licensed Java ME—and no evidence that Danger licensed Java SE, as Oracle now contends. JA436; 2016 Tr. 1671; JA678-692 (Danger-Sun licensing agreement). *Oracle* may view SavaJe and Danger’s implementations as proof that Android competed with Java SE in smartphones, but the jury was entitled to disagree. *E.g.*, TX7803 at 5-6 (Oracle recognition that “Java, as a mobile platform, [is] not competitive”).

Finally, Oracle misconstrues the trial record in contending that Amazon “ping-ponged between Java and Android” for the “Kindle.” Oracle Br. 11, 15. Oracle conflates two different Amazon products and two different versions of Java. The “Kindle” is an electronic reader (lacking various smartphone capabilities) and in Oracle’s own words, “Amazon licensed Java ME . . . for its electronic reader, Kindle.” Oracle 13-1021 C.A.

Resp. & Reply Br. 56; *see also* 2012 Tr. 3371. In contrast, the “Kindle Fire,” a multi-purpose tablet, is “both a device and an operating system” “built on Android.” JA206; *see* JA395-397. There is *no* evidence that Amazon “ping-ponged” between Java *SE* and Android. Amazon licensed Java *ME* for one product and used Android for a very different product.

3. *The Jury Was Entitled to Find No Market Harm to Java SE.*

Oracle’s primary argument is that Google’s reuse cannot be fair use because Android competes with Java SE. Oracle’s premise and conclusion are both wrong.

Initially, Oracle errs in suggesting that creating a better product weighs against a finding of fair use in the software context. Because the fair-use doctrine must be adapted to the nature of the copyrighted material, Oracle’s literary analogies fall short. The jury’s verdict does not threaten Oracle’s amici that create traditional literary and musical works, including the Association of American Publishers and the Recording Industry Association of America. Copyright does not foster literary creativity by encouraging development of better and more efficient versions of a novel; but encouraging innovation in software requires fostering exactly those kinds of improvements. “When technological change has rendered its literal terms ambiguous, the Copyright Act must be construed in light of [its] basic purpose” of “promoting broad public availability” of new creative works. *Sony Corp. of Am. v. Universal City Studios, Inc.*, 464 U.S. 417, 432 (1984) (quoting *Twentieth Century Music Corp. v. Aiken*, 422 U.S. 151, 156 (1975)). The United States Copyright

Office has explained that, “[i]n the context of software-enabled consumer products in particular, the fourth factor is likely to favor fair use where the purpose of the use is to create a ‘legitimate competitor in the market.’” U.S. Copyright Office, *Software-Enabled Consumer Products* 58-59 (Dec. 2016) (quoting *Sony*, 203 F.3d at 607). Oracle’s contrary view would allow a copyright owner to block innovation and stifle competition by preventing innovators from reusing a small amount of functional code in a new larger work that has enormous creative value.

In any event, the record contains abundant evidence that Android did not compete with Java SE or displace it in any existing market, as explained above. Oracle’s brief fails to identify a *single* lost Java SE sale. At trial, its “market-harm” expert admitted he did not “know whether” “Java SE” “revenue” is “going up or down,” and that he did not “go to the trouble of talking to the Java SE managers to see how [it] was doing.” JA674-675. Oracle’s witnesses testified that Java SE was “doing fine,” with growing revenues. JA675. Google’s market-harm expert testified that “Android does not have any market impact on Java SE,” and that “Android has not superseded Java SE” in the marketplace. JA255-256. The jury was entitled to credit that evidence.

Oracle’s assertion (Oracle Br. 47) that the Federal Circuit “cited only *undisputed* evidence in finding that Oracle suffered harm to potential markets” is puzzling—because the Federal Circuit cited literally *no evidence* other than the failed partnership negotiations between Oracle and Google in finding “that Oracle intended to license Java SE in smartphones.” Pet. App. 51a-52a. “The market for potential derivative

uses includes only those that creators of original works would in general develop or license others to develop.” *Campbell*, 510 U.S. at 592. But Oracle relies only on speculative markets rather than addressing “any part of the normal market for” Java SE. *Harper & Row*, 471 U.S. at 568 (quoting S. Rep. No. 94-473, at 65).

The jury heard substantial evidence that there was *no* potential market for Java SE in smartphones because Java SE is not suitable for modern smartphones. Google Br. 43, 48; TX7362 at 1 (Oracle employee stating that “[w]e have no solution for smart phones, true”). The jury heard that Oracle gave up on making its own smartphone platform or adapting Java SE to work in smartphones. *E.g.*, JA102; JA234-235; JA650. And Oracle’s brief fails to identify a single company that ever expressed interest in licensing Java SE for a modern smartphone. Any potential future market would have been a market for a materially different product than Java SE. If a hypothetical, wished-for market for a different product were enough to find market harm, that factor would never support fair use.

In sum, there is no basis on this record to overturn the jury’s verdict that Google’s reuse of the Java SE declarations was fair use—a verdict that is fully consistent with the application of copyright law to further interoperability and innovation in computer software.

CONCLUSION

The judgment of the court of appeals should be reversed.

Respectfully submitted,

Kent Walker
Catherine Lacavera
Renny Hwang
GOOGLE LLC
1600 Amphitheatre Parkway
Mountain View, CA 94043

Lisa S. Blatt
David M. Krinsky
Sarah M. Harris
Meng Jia Yang
WILLIAMS & CONNOLLY LLP
725 Twelfth Street, N.W.
Washington, DC 20005

Robert A. Van Nest
Christa M. Anderson
Eugene M. Paige
Reid P. Mullen
KEKER, VAN NEST
& PETERS LLP
633 Battery Street
San Francisco, CA 94111

Thomas C. Goldstein
Counsel of Record
Sarah E. Harrington
Kevin K. Russell
Daniel Woofter
Erica Oleszczuk Evans
GOLDSTEIN & RUSSELL, P.C.
7475 Wisconsin Avenue
Suite 850
Bethesda, MD 20814
(202) 362-0636
tg@goldsteinrussell.com

Michael S. Kwun
KWUN BHANSALI LAZARUS LLP
555 Montgomery Street
Suite 750
San Francisco, CA 94111

Bruce W. Baber
Marisa C. Maleck
KING & SPALDING LLP
1180 Peachtree Street, N.E.
Atlanta, GA 30309

March 11, 2020