No. 14-410

IN THE

Supreme Court of the United States

GOOGLE INC.,

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 IN OPPOSITION

DORIAN E. DALEY
DEBORAH K. MILLER
ORACLE AMERICA, INC.
500 Oracle Parkway
Redwood Shores, CA 94065

DALE M. CENDALI
KIRKLAND & ELLIS LLP
601 Lexington Avenue
New York, NY 10022

E. JOSHUA ROSENKRANZ
Counsel of Record
MARK S. DAVIES
ANNETTE L. HURST
ORRICK, HERRINGTON & SUTCLIFFE LLP
51 West 52nd Street
New York, NY 10019
(212) 506-5000
jrosenkranz@orrick.com

Counsel for Respondent

QUESTION PRESENTED

The Copyright Act protects “original works of authorship,” including “literary works,” 17 U.S.C. § 102(a), which Congress defined as “works … expressed in words, numbers, or other verbal or numerical symbols or indicia,” *id.* § 101. “Computer program[s],” also defined by the Copyright Act, are literary works. *Id.* Google copied at least 7000 lines of original computer source code that Oracle wrote, and included the copied code in its own software platform, even though Google could have written its own code (rather than copying Oracle’s) to perform the same functions. The question presented is:

Does the Copyright Act protect Oracle’s computer source code that Google copied, where Google concedes that the code was original and creative, and Oracle could have written its code in any number of ways to perform the same function?

CORPORATE DISCLOSURE STATEMENT

Respondent in this Court, plaintiff-appellant below, is Oracle America, Inc. Oracle America, Inc. is not publicly traded. It is a subsidiary of Oracle Corporation (NYSE: ORCL), a publicly traded company. No other publicly held company owns 10% or more of its stock.

TABLE OF CONTENTS

Page(s)

TABLE OF AUTHORITIES

Page(s)

Cases

Kennedy v. Plan Adm'r for DuPont Sav. & Inv. Plan,
555 U.S. 285 (2009)..36

Kepner-Tregoe, Inc. v. Leadership Software, Inc.,
12 F.3d 527 (5th Cir. 1994)...........................21, 22

KSR Int'l Co. v. Teleflex, Inc.,
550 U.S. 398 (2007)..30

Lexmark Int'l, Inc. v. Static Control Components, Inc.,
387 F.3d 522 (6th Cir. 2004).............14, 15, 24, 25

Lotus Dev. Corp. v. Borland Int'l, Inc.,
49 F.3d 807 (1st Cir. 1995), *aff'd*,
516 U.S. 233 (1996)...........2, 11, 13, 14, 19, 20, 37

Mazer v. Stein,
347 U.S. 201 (1954)......................21, 25, 29, 30, 31

Mitel, Inc. v. Iqtel, Inc.,
124 F.3d 1366 (10th Cir. 1997)...........................19

Sega Enters. Ltd. v. Accolade, Inc.,
977 F.2d 1510 (9th Cir. 1993).............................37

Sid & Marty Krofft Television Prods., Inc. v. McDonald's Corp.,
562 F.2d 1157 (9th Cir. 1977).............................21

Situation Mgmt. Systems, Inc. v. ASP Consulting Group,
560 F.3d 53 (1st Cir. 2009) 20

Sony Corp. of Am. v. Universal City Studios, Inc., 464 U.S. 417 (1984) 31

Twentieth Century Music Corp. v. Aiken, 422 U.S. 151 (1975) 31

Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc., 797 F.2d 1222 (3d Cir. 1986) 21, 23

Constitutional Provisions

U.S. Const. Article I, § 8, cl. 8 31

Federal Statutes

17 U.S.C. § 101 .... 12, 16, 28

17 U.S.C. § 102 .... *passim*

17 U.S.C. § 106 31

17 U.S.C. § 107 37

17 U.S.C. § 302 29

Legislative History

H.R. Rep. No. 1476-94 (1976), *reprinted in* 1976 U.S.C.C.A.N. 5659 .... 12, 27, 28, 29

S. Rep. No. 473-94 (1975), *available at* 1975 WL 370212 .... 27, 28, 29

Other Authorities

INTRODUCTION[1]

The court of appeals' decision is a careful and sound application of settled legal doctrine to an outrageous set of facts. Oracle spent years and millions of dollars writing packages of software code and nurturing a fan base of millions of programmers attracted to the code's elegance. Google then copied thousands of lines of that code into its Android platform for the express purpose of capturing Oracle's fan base. The code Google copied was the very code that the fan base liked the most because its elegant expression was so easy to remember and use. What Google did was the equivalent of plagiarizing the topic sentence of every paragraph of a blockbuster novel, as well as the chapter and subchapter titles, and then paraphrasing the rest. Google's argument—that the code lost all copyright protection because it became popular and Google wanted Android to be popular, too—would not fly for any other work. It does not fly for computer code.

The court of appeals found that Oracle's code far exceeded the minimum level of originality necessary to make it eligible for copyright protection because—as Google conceded—the code was original and highly creative. There is nothing remarkable about that ruling. Every circuit to address the question agrees that computer code this extensive and creative is protectable. Google does not cite a single

[1] The Joint Appendix below is "C.A." Documents preceded by "C.A." were filed in the court of appeals. Amicus briefs are cited as "___ Br.," according to the lead amicus's name or abbreviation. All emphasis added unless noted.

case—literally none—that has ever held that code so extensive is devoid of copyright protection.

The main thrust of Google's petition is that this case presents "the [same] question," Pet. 1, that the Court considered 20 years ago in *Lotus Development Corp. v. Borland International, Inc.*, 49 F.3d 807 (1st Cir. 1995), but never resolved, 516 U.S. 233 (1996). Based on a single sentence in *Lotus* (quoted *infra* 19), Google poses the question whether the lines of code Oracle authored constitute a "method of operation"—and are therefore devoid of all copyright protection—under § 102(b) of the Copyright Act. But *Lotus* did not present the same question Google poses because, as *Lotus* states explicitly, it is not about computer code. Today, no circuit—including the First—applies the standard Google proposes. In any case, Google never argued to the court of appeals that the lines of code are a "system" or "method of operation."

To hold unprotectable the thousands of lines of code Google copied would strip all code of copyright protection. All code can be described exactly as Google describes Oracle's: code "use[d] … to cause a computer to perform certain functions." Pet. 2. The very suggestion that Oracle's code loses copyright protection would devastate the software industry. Copyright protection is central to innovation in the software industry. There is no dispute that Google was free to write its own code to perform the same functions as Oracle's. Instead, it plagiarized.

Precisely because every circuit agrees that code like Oracle's is eligible for copyright protection, the decision below maintains the copyright regime that has fostered software innovation. Verbatim copying of another's original code threatens the activity at the core of today's economy. The law is clear and simple: *Write your own code*. Or, if you want to harness another author's fan base for profit, take a license—which Oracle offered and Google rejected. Theft is theft whether the work is a novel or complex software.

That is not to say that no one can ever copy. There are circumstances when copying is fair use. That doctrine accommodates the various policy concerns Google and its amici raise, such as compatibility and lock-in. Google is free to try to prove on remand that its copying was fair, even though "Google designed Android so that it would *not* be compatible with the Java platform." App. 56. If Google's fair-use defense fails, it can seek review again. But there is nothing cert-worthy about an argument that every one of Oracle's thousands of lines of code lost copyright protection just because that code became wildly popular.

The petition should be denied.

STATEMENT OF THE CASE

Oracle Develops Computer Programs That Help Programmers Write Their Own Apps

Sun Microsystems, now part of Oracle, "developed" the Java programming language and "the Java 'platform.'"[2] App. 4. They are two different things. The Java language is a computer programming language made up of "words, symbols, and other units, together with syntax rules for" programming. *Id.* The Java platform was designed to "relieve programmers from the burden of writing different versions of their computer programs for different operating systems or devices." *Id.* With the Java platform, "a software programmer could 'write [programs] once, [that could] run anywhere.'" *Id.* "Write once, run anywhere" became Java's credo.

The most important component of the platform here is its thousands of prewritten programs. Oracle wrote those "ready-to-use Java programs to perform common computer functions and organized those programs into groups it called 'packages.'" App. 5.[3] The packages "allow programmers to use the prewritten code to build certain functions into their own programs, rather than writ[ing] their own code to perform those functions from scratch." *Id.*

For example, in one package, Oracle wrote a program called "URLConnection" to establish an internet connection to a website. C.A. 10,013-28, 20,753-54. As simple as that sounds, it is exceedingly

[2] "Oracle acquired Sun in 2010," App. 4 n.1, and "owns the copyright on [the] Java [platform] and the API packages," App. 7. For simplicity, we refer to Oracle to cover creative efforts that Sun began and Oracle continued.

[3] These software packages are sometimes called "APIs" ("application programming interfaces"). The term "API" can be highly imprecise. Like the opinion below, we refer more precisely to "packages" of software. *Id.*

complex. Developers needed special expertise to write the network protocols and cryptographic algorithms. Once they did, later programmers using the Java language could either “re-invent[] the wheel[],” by writing new algorithms, App. 117, or save huge amounts of time by using the code Oracle wrote. If they chose the latter and wished to write a program that connects to BankofAmerica.com, they would call on Oracle’s prewritten code by typing (or “declaring”):

```
new URL(‘http://www.bankofamerica.com’)
.open Connection().
```

If programmers used the feature frequently, they would not bother looking up what to call it, since it would become second nature. C.A. 20,937-38.

“[D]esigning the Java API packages [i]s a creative process.” App. 22. It took Oracle’s “most senior[,] experienced and talented” developers years to write some of the packages. C.A. 20,459; *see* C.A. 20,791, 20,921. The developers worked with a “clean slate,” C.A. 21,412, 20,913, meaning when they began writing the packages, they “had unlimited options,” App. 33 (quotation marks omitted).

The Java packages are comprised of Java “source code”—code written “in a human-readable language.” App. 5 (quotation marks omitted). Every Java package is arranged in an intricate hierarchy and divided into related source code files called “classes” and “interfaces.” App. 5-6. Those classes and interfaces, in turn, contain numerous “methods,” each of which is a computer program that performs a discrete function, like opening an internet connection.

Id. Because no particular groupings are required by Java, C.A. 20,788, Oracle's developers undertook many "creative choices … in designing a Java package." App. 33 n.6.

Every class and method has "two types of source code[,] … declaring code[] and … implementing code." App. 6. The "declaring code" is composed of various elements referred to as declarations, headers, signatures, and names. App. 6-7. The declaring code is the opening line or lines of a program in the package, such as

> *public URLConnection openConnection() throws java.io.IOException.*

These lines specify the program's "inputs, name[,] and other functionality." App. 7. The code can—but does not always—reflect the component's place in the package hierarchy. App. 104, 110-114. The implementing code "gives the computer the step-by-step instructions for carrying out the declared function," such as opening the internet connection. App. 7.

In that way, declaring code is like the topic sentences and chapter and subchapter headings of the Java platform, whereas the implementing code is the body of each paragraph following the topic sentence. The difference is that programmers never see the implementing code. They need to know only the declaring code to call on Java's prewritten programs.

As evident from this description, programmers *do not have to* invoke these packages to write programs in the Java language. C.A. 20,458-59. With only a few minor exceptions (portions of "three …

packages"), programmers can write in the Java language without using the packages. App. 6.

Oracle invested hundreds of millions of dollars in developing the Java platform, C.A. 20,454, 20,557, and registered the platform with the Copyright Office, C.A. 1066. Google admits the declaring code is highly creative as were the structural and organizational decisions of what programs to group together and how they would interrelate. App. 37 ("Google's own 'Java guru' conceded that there can be 'creativity and artistry even in a single method declaration.'" (quoting C.A. 20,970)); *see also* C.A. 20,761-66, 21,949 (Oracle's chief Java architect). Part of what made the platform so popular was how graceful and elegant it is to everyday programmers. C.A. 20,937-38.

Although Oracle owns the copyright on the platform and its packages, Oracle actively encourages others to use them through a carefully calibrated licensing regime. App. 8. Oracle nurtured a community of programmers and taught them how to use those packages to develop their own applications (commonly known as "apps"). C.A. 20,557, 21,438-39. Oracle offers a free "'open source' license," which allows programmers to use the software packages so long as they "contribute back" their advances. App. 8. For those who want to develop proprietary work and not dedicate it back to the Java community, Oracle offers two license options—including an option for businesses, like Google, that want to use the declaring code but write their own implementing code. *Id.* Oracle granted licenses to "just about every smart phone carrier … around the world," including to its competitors. C.A. 22,237. All these licenses were conditioned on the developer's commitment to write code compatible with the Java platform. Accordingly, Oracle offered a free open-source license while retaining the ability to earn "valuable" and "very lucrative" licensing revenue from those who licensed Oracle's packages for their own profit. *Id.* No one thought they could copy Oracle's code and use it commercially without a license.

Without A License, Google Copies Thousands Of Lines Of Computer Source Code

When designing its Android platform, Google wanted to "leverag[e] Java for its existing base of developers" to develop apps for Android so Android would be more attractive to consumers. App. 57. Google identified 37 software packages it "believed Java application programmers would want" in Android. App. 9 (quotation marks omitted).

Google executives knew Google "[m]ust take [a] license from [Oracle]," C.A. 1132, because it "own[s] the … IP," C.A. 1200-01. Oracle was willing to negotiate a license, but Google declined, "refus[ing]" to accede to the one condition imposed on every commercial licensee: to make Android "compatible with the Java [platform]" and "interoperable with other Java programs." App. 9.

Instead, without a license, Google "copied the declaring source code from the 37 … packages verbatim, inserting that code into parts of its Android software." App. 9-10. Across the 37 packages, Google copied the declaring code for thousands of classes, interfaces, and methods, amounting to at least "7,000 lines of declaring code." App. 14-15.

"Google released the Android platform in 2007." App. 10 "[A]ll Android phones contain copies of the accused … software." *Id.* "Google provides the Android platform free of charge to smartphone manufacturers." App. 11. This was no act of altruism. As Google's former CEO testified: "The vast majority of Google's revenue … comes from search revenue. And so the primary reason to have … Android is that people will do more searches, and then we'll get more money …." C.A. 21,631. Google's copying made it impossible for Oracle to compete by licensing its platform in the then-nascent smartphone market. As Oracle's president observed, "It's … hard to compete with free." C.A. 22,498.

The District Court Rules Oracle's Original And Creative Source Code Not Copyrightable

Oracle sued Google for copyright infringement. App. 2. "[T]he parties agreed that the jury would decide infringement [and] fair use, … while the district judge would decide copyrightability"—i.e., whether what Google copied was eligible for copyright protection. *Id.* "[T]he jury returned a verdict finding that Google infringed Oracle's copyright in the 37 Java API packages," but "hung" on fair use. App. 12.

There was no fair-use retrial, however, because the district court concluded that what Google copied was devoid of copyright protection. App. 164. The court conceded that the declaring code contains "creative elements," App. 104, but held that "no matter how creative or imaginative," the ideas in the code merged with its expression, precluding protection. App. 155. The court also held the packages' "creative" and "original" structure and organization unprotected as “a command structure for a system or method of operation.” App. 158-59 (citing 17 U.S.C. § 102(b)).

The Court Of Appeals Holds That Oracle’s Packages Are Copyrightable

The court of appeals reversed. App. 3.

The court observed that the “Copyright Act provides protection to ‘original works of authorship.’” App. 17 (quoting 17 U.S.C. § 102(a)). “‘Original, as the term is used in copyright, means only that the work was independently created by the author (as opposed to copied from other works), and that it possesses at least some minimal degree of creativity.’” App. 18 (quoting *Feist Publ’ns, Inc. v. Rural Tel. Serv. Co., Inc.*, 499 U.S. 340, 345 (1991)). “It is well established that copyright protection [for computer programs] can extend to” their “source code” and to their “structure[] and organization.” App. 20-21.

Applying these principles, the court found it “undisputed that Google copied 7,000 lines of declaring code and generally replicated the overall structure, sequence, and organization of Oracle’s 37 … packages.” App. 14-15. “[I]t is [also] undisputed that the declaring code and the structure and organization of the … packages are original.” App. 22.

The court rejected Google’s argument that the source code is ineligible for copyright protection because its original expression “merge[d]” with unprotectable ideas. App. 30. There could be “no merger” of the code’s expression with its ideas because Oracle’s developers “had unlimited options as to the selection … of the 7,000 lines Google copied." App. 33-34 (quotation marks omitted).

Apart from the declaring code, the court concluded that the packages' intricate structure and organization merited copyright protection. Below, but not in this Court, Google argued that the packages' structure and organization is an unprotectable method of operation. The court of appeals observed that this argument "relie[s] upon" a lone sentence in *Lotus*. App. 41. Taken out of context, that solitary sentence, which "cite[s] no authority" in support, seems to define an unprotectable "'method of operation'" as the "'means by which a person operates something, whether it be a car, a food processor, or a computer.'" App. 42 & n.10 (quoting *Lotus*, 49 F.3d at 815). The court found *Lotus* "distinguishable on its facts"—for three reasons discussed at length below (at 13, 19). App. 43. It also held *Lotus*'s one sentence "inconsistent with Ninth Circuit law," *id.*, and that "no other circuit has adopted the First Circuit's 'method of operation' analysis,'" App. 45.

The court then remanded for retrial on fair use. App. 58-70.

REASONS TO DENY CERTIORARI

The petition should be denied because: (I) every circuit agrees that copyright protects such an extensive amount of original source code; (II) there is no circuit disagreement implicated on any relevant dispositive issue; (III) the decision below is compelled by the Copyright Act's plain language and this Court's precedents; and (IV) this interlocutory appeal is not an appropriate vehicle for deciding this copyright question.

I. COURTS AGREE THAT THOUSANDS OF LINES OF ORIGINAL SOURCE CODE ARE COPYRIGHTABLE

Google's lead argument is that "Courts … Are In Disarray About The Application Of Section 102(b) To Software." Pet. 13. This Court should deny certiorari because no circuit disagrees that 7000 lines of original and creative code are copyrightable.

1. The statutory analysis is simple: "Copyright protection subsists … in original works of authorship … includ[ing] … literary works." 17 U.S.C. § 102(a). "Literary works" are works "expressed in *words*, *numbers*, or other verbal or numerical *symbols*." *Id.* § 101. "Computer programs," also defined in § 101, are "literary works" and copyright protected "'to the extent that they incorporate authorship in the programmer's expression of original ideas, as distinguished from the ideas themselves.'" App. 17-18 (quoting H.R. Rep. No. 1476-94 at 54 (1976), *reprinted in* 1976 U.S.C.C.A.N. 5659, 5667) ("H.R. Rep.").

As the court of appeals observed, App. 20, every circuit to consider the issue recognizes as "well established" that computer source code is copyrightable as long as it is original. *Computer Assocs. Int'l, Inc. v. Altai, Inc.*, 982 F.2d 693, 702 (2d Cir. 1992); *accord Gen. Universal Sys., Inc. v. Lee*, 379 F.3d 131, 142 (5th Cir. 2004); *Autoskill Inc. v. Nat'l Educ. Support Sys., Inc.*, 994 F.2d 1476, 1492 n.18 (10th Cir. 1993); *Johnson Controls, Inc. v. Phoenix Control Sys., Inc.*, 886 F.2d 1173, 1175 (9th Cir. 1989); *Apple*

Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1247-49 (3d Cir. 1983).

Below, "Google nowhere dispute[d] that [legal] premise." App. 21. Nor do its amici—several of which confirm that original "computer program code deserves copyright protection." H.P. Br. 3; *accord* Computer Scientists Br. 2.

2. Google does not point to any opinion holding—or even suggesting—that a complex array of thousands of lines of code deserves no copyright protection. In fact, the two cases Google highlights in trying to demonstrate a circuit conflict, *Lotus* and *Lexmark*, confirm that original source code is protectable.

***Lotus* (1st Cir.)**. *Lotus* is a strange case to feature for the proposition that there is "disarray" over code, Pet. 13, because *Lotus* was not about the copyrightability of code and even expressly found Lotus's code copyrightable and not a method of operation. The issue in *Lotus* was that the defendant copied the menu command hierarchy from the Lotus 1-2-3 spreadsheet program, consisting of rudimentary commands "such as 'Copy,' 'Print,' and 'Quit.'" 49 F.3d at 809. In deciding whether this "menu command hierarchy is copyrightable," the First Circuit repeatedly emphasized that the defendant's program "did not copy any of Lotus's underlying computer code; it copied only the words and structure of Lotus's menu command hierarchy." *Id.* at 809-10; *accord id.* at 812. Though "the Lotus 1-2-3 code [wa]s not before [the First Circuit]," *Lotus* agreed with all the other courts that "original computer codes generally are protected by copyright." *Id.*

at 816 n.11. When concluding that the menu command hierarchy was uncopyrightable, *Lotus* was clear that its holding did *not* apply to computer code:

> The Lotus menu command hierarchy is also *different* from *the underlying computer code* because while code is necessary for the program to work, its precise formulation is not. In other words, to offer the same capabilities as Lotus 1-2-3, [the defendant] did not have to copy Lotus's underlying code (and indeed it did not) …. *Thus the Lotus 1-2-3 code is not a[n] uncopyrightable "method of operation."*

Id. at 816. Thus, under *Lotus*, code is copyrightable as long as it is sufficiently complex and creative to be original. *Accord Hutchins v. Zoll Med. Corp.*, 492 F.3d 1377, 1383 (Fed. Cir. 2007) (applying First Circuit law) ("The Copyright Act['s] … protection against unauthorized copying of computer programs … applie[s] to protect computer codes and design and text ….").

***Lexmark* (6th Cir.)**. The Sixth Circuit's opinion in *Lexmark International, Inc. v. Static Control Components, Inc.*, 387 F.3d 522 (2004), is also fully consistent with the opinion below. *Lexmark* agrees that "computer programs may be entitled to copyright protection … with respect to … [their] source code." 387 F.3d at 533. In agreeing with the Third Circuit's protection of Apple's operating system programs in *Apple*, 714 F.2d 1240, the Sixth Circuit left no doubt it would hold protectable 7000 lines of original code. *Lexmark*, 387 F.3d at 539. The key, the Sixth Circuit emphasized, is whether the code is original expression, i.e., whether the initial author had "alternative means of expression": "Given the nature of the Apple program[s], it would have been exceedingly difficult to say that practical alternative means of expression did not exist" thereby rendering the programs uncopyrightable. *Id.*

The Sixth Circuit reached the opposite conclusion in *Lexmark* because the work there was not creative or original. It was a snippet of code that occupied merely "55 bytes of memory"—less than "the phrase 'Lexmark International, Inc. vs. Static Control Components, Inc.'" *Id.* at 529-30. *Lexmark* held this snippet not original and thus not protectable because it was "dictate[d]" by "external constraints," leaving the programmer without "much choice" in the code. *Id.* at 539-40 (quotation marks omitted). The "size and complexity" of the *Lexmark* program was a "lamppost" compared to "the Sears Tower" that was Apple's program. *Id.* at 539.[4]

Lotus certainly has not created "disarray" about the copyrightability of computer code. Pet. 13. Since *Lotus*, only one certiorari petition has addressed computer copyright. Petition for Writ of Certiorari, *Softel, Inc. v. Dragon Med. & Scientific Commc'ns, Inc.*, (No. 97-999), 1997 WL 33548900 (S. Ct. Dec. 15, 1997), *cert. denied* 523 U.S. 1020 (1998).

3. Given the unanimity of the above-cited authorities, Google cannot contend that original code

[4] *ATC Distribution Group, Inc. v. Whatever It Takes Transmissions & Parts, Inc.*, 402 F.3d 700 (6th Cir. 2005) (cited at Pet. 16), is not a case about computer code.

lacks copyright protection. Instead, Google's argument must be that there is something special about *Oracle's* code that precludes copyright protection. But, on this narrower proposition, Google proffers no case where code is original yet nevertheless ineligible for copyright protection.

According to Google, Oracle's thousands of lines of code, though original, lost all copyright protection because they "constitute, or embody, a system or method of operating the pre-written programs." Pet. 27. But *all* computer code is "a system or method of operating … programs." *Id.* The Copyright Act defines "computer program[s]" as "set[s] of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result." 17 U.S.C. § 101. If *this* code loses copyright protection, then all code does, which is contrary to Congress's copyright protection of computer code, as all the circuits recognize.

Google does not overcome these difficulties by labeling the thousands of lines of code "basic shorthand commands of the Java language," or "the basic vocabulary words of the Java language." Pet. 2 (without citation). For one thing, that is false. The declaring code and packages are *not* part of the Java language; programmers can program in the Java language without Oracle's declaring code or packages. *Supra* 4; App. 4, 6.[5]

[5] Google's argument seems to be part of a strategy to conflate the Java language with the Java platform. Pet. 5, 29,

Labeling Oracle's concededly creative and original work "shorthands" does not mean the work is ineligible for copyright protection. Perhaps Google means that some lines of code are short, when considered individually. But that does not mean that the whole work loses copyright protection. App. 36-37. Moreover, many lines of declaring code are not short:

```
public abstract void verify (PublicKey
key, String sigProvider)
  throws CertificateException, NoSuchAlgorithmException,
  InvalidKeyException,
  NoSuchProviderException, SignatureException
```

C.A. 10,042. Maybe Google means that this code can be used as a shortcut to call other code. Pet. 6-7, 29-30. But that, too, is a common trait of computer code and is irrelevant to copyrightability.

Mainly, Google uses "shorthand" to analogize this case to the system for shortening words in *Brief English Systems, Inc. v. Owen*, 48 F.2d 555 (2d Cir. 1931). Pet. 29-30. This 1931 case involved no source code. Moreover, that plaintiff was not trying to protect any actual literary work; it was trying to use copyright to protect its idea of a system for shortening words (e.g., by eliminating vowels) and thus prevent others from using that system to shorten words. 48 F.2d at 556.

32, 35. But Oracle has not asserted a copyright infringement claim as to the Java language. App. 48.

Oracle seeks no such protection. Anyone is free to write software that performs the same functions and embodies the same ideas as Oracle's. App. 47-49. What no one (including Google) is free to do is to copy Oracle's declaring code, "the precise strings of code [it] used." App. 48 (quotation marks omitted). Google concedes Oracle's code is original, and all courts agree that original code is copyrightable.

II. NO CIRCUIT SPLIT IS IMPLICATED

With no circuit split regarding code, Google tries to demonstrate that the court of appeals' basic approach is inconsistent with other circuits. Google claims three splits: (A) what courts treat as an unprotectable "method of operation" under § 102(b); (B) application of the merger doctrine; and (C) treatment of compatibility. Google is mistaken.

A. There Is No Split On "System" Or "Method Of Operation"

The "system" and "method of operation" nomenclature is found in 17 U.S.C. § 102(b):

> In no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work.

Google claims courts are "deeply divided" on what is a "method of operation," and cites *Lotus* as its primary support. Pet. 13-14. This is not so.

The court below offered three grounds to distinguish *Lotus*, meaning this case does not split from *Lotus*. First, *Lotus* did not involve computer code. *Supra* 13.

Second, *Lotus* "found that the commands at issue there (copy, print, etc.) were not creative." App. 43. In contrast, "it is undisputed here that the declaring code and the structure and organization of the API packages are both creative and original." *Id.*

Third, *Lotus* "found the commands at issue were 'essential to operating' the system." App. 43. *Lotus* explained that because "the Lotus menu commands," like the buttons on a VCR, "are *essential* to operating Lotus 1-2-3," the menu commands are not copyrightable. 49 F.3d at 817. In contrast, Google could have used the Java language without duplicating Oracle's code or the packages' structure. App. 43. "[I]t is undisputed that … Google did not need to copy the structure, sequence, and organization of the Java API packages to write programs in the Java language." App. 43; *accord* App. 57. Google could have changed the "method and class names … and [they] still [would] have worked." App. 103-04, 117.

Moreover, there is no split—only disagreement about one opaque and unsupported *sentence* in *Lotus* that the First Circuit itself does not follow. *See Mitel, Inc. v. Iqtel, Inc.*, 124 F.3d 1366, 1372 (10th Cir. 1997). The sentence reads: "We think that 'method of operation' as that term is used in § 102(b), 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 First Circuit could not have meant that an otherwise protected work loses all protection whenever it is *also* "the means by which a user operates something." *Id.* at 816. That would render all computer code unprotected, *supra* 16, and is inconsistent with the statute's language and this Court's precedent, *infra* 24-31.

The First Circuit has since clarified that it does not interpret § 102(b) as Google asserts. In *Situation Management Systems, Inc. v. ASP Consulting Group*, 560 F.3d 53, 55 (2009), the First Circuit held that some instructional materials were copyrightable even though they contained "system[s]" and "process[es]" "for effective communication and negotiation within the workplace" that were not copyrightable. The court explained that the particular "*descriptions* of a process or system are copyrightable," "including the works' overall arrangement and structure." *Id.* at 61 (emphasis in original). Section 102(b) merely means that the copyright protection does not "extend" beyond the author's words used to describe the process such that the author claims to own "the underlying process or system itself." *Id.*

No court has adopted *Lotus*'s one sentence as a description of how § 102(b) works. App. 45. So, rather than "deepen[ing]," Pet. 13, any purported split, if ever there was one, has evaporated.

B. The Courts Do Not Disagree About Merger

Google asserts that courts disagree on how to apply the idea/expression dichotomy and merger doctrine that arises from it. Pet. 16-18. The idea/expression dichotomy recognizes that "[copyright] protection is given only to the expression of the idea," "not [to] the idea itself." *Mazer v. Stein*, 347 U.S. 201, 217 (1954). Merger holds: "When the idea and its expression are … inseparable," because there is only one way to express that idea, "copying the expression will not be barred, since protecting the expression in such circumstances would confer a monopoly of the 'idea.'" *Sid & Marty Krofft Television Prods., Inc. v. McDonald's Corp.*, 562 F.2d 1157, 1168 (9th Cir. 1977) (citation omitted); *accord* App. 30. If there is only one way to write a program, then the idea merges into the expression and the program is not protectable.

There is no split on how this principle applies to computer code. Consistent with this Court's protection of a work's original expression, the circuits uniformly recognize that copyright subsists in a computer program's particular expression when it is not the only way to achieve the program's purpose or carry out the program's function. App. 21, 47-48; *see Kepner-Tregoe, Inc. v. Leadership Software, Inc.*, 12 F.3d 527, 533-34, 536 (5th Cir. 1994); *Altai*, 982 F.2d at 701-03; *Atari Games Corp. v. Nintendo of Am., Inc.*, 975 F.2d 832, 839 (Fed. Cir. 1992); *Johnson Controls*, 886 F.2d at 1175-76; *Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc.*, 797 F.2d 1222, 1233-39 (3d Cir. 1986).

In *Atari*, Nintendo developed a program that made it impossible for a game to work without transmission of the correct coded message. 975 F.2d at 836. The court found that "Nintendo … exercised creativity in the selection and arrangement of its instruction lines." *Id.* at 840. Because there were "alternate" ways of achieving the same result—a security program—Nintendo was able to protect its expression embodied in the "creativity [of] the [code's] selection and arrangement," as distinct from the program's "process or method." *Id.* at 839-40. Similarly, in *Johnson Controls*, the court found copyright protected the structure and organization of a program "to control wastewater treatment plants." 886 F.2d at 1174. That the program was often "adapt[ed]" for "individual application[s]" demonstrated that there was "room for individualized expression" as well as "discretion and opportunity for creativity" in the program. Accordingly, the program was original and did not merge with its function. *Id.* at 1176.

That is exactly what the court of appeals held here. App. 32-35, 46-49. It was undisputed that "Oracle had unlimited options as to the selection and arrangement of the 7,000 lines [of code] Google copied." App. 33 (quotation marks omitted); *accord* App. 34, 48-49. Because this is *not* a case where Oracle "had only one way, or a limited number of ways, to write [the declaring code]," the court held that Oracle's expression in its code and in the packages' organization did not merge. App. 32-33; *accord* App. 104, 117, 157-58.

Instead of identifying a split, Google points to modest differences in how various courts describe their process and allocate the burden of proof. Pet. 16-19. Most courts use the "Abstraction-Filtration-Comparison" test. *See Gates Rubber Co. v. Bando Chem. Indus., Ltd.*, 9 F.3d 823, 834-39 (10th Cir. 1993); *Altai*, 982 F.2d at 706-11; App. 23-25. Others use different phrasing. *Kepner-Tregoe*, 12 F.3d at 534; *Johnson Controls*, 886 F.2d at 1175-76;

Whelan, 797 F.2d at 1234-37; *see Gates Rubber*, 9 F.3d at 840 ("*Whelan* is premised upon traditional principles of copyright law."). But these courts all ask the same question: Does the program's expression merge with its purposes and functions? In each case, the court conducts an extensive analysis—as the court did here, App. 32-35, 46-49—endeavoring to separate protectable expression from unprotectable ideas and functions. In light of Google's concessions and the undisputed record, App. 33-34, 48-49, Google cannot point to how these various articulations yield different results in this case, or any other.

C. This Case Presents No Cert-Worthy Split About Compatibility

Google claims courts are "divided on … the relevance of compatibility to copyrightability." Pet. 18. As the court of appeals observed, any such disagreement is "[i]rrelevant," App. 50, because "the record evidence [shows] that Google designed Android so that it would *not* be compatible with the Java platform." App. 56.

Platforms are not compatible unless programs written for one run on the other. C.A. 22,348 (Google's expert); App. 159-60 (district court). Though "Google repeatedly cites to the district court's finding" of partial interoperability—a contradiction in terms that Google repeats here (Pet. 10, 31)—"Google … cites to no evidence" "to support [its] claim" of compatibility. App. 56 & n.15. The record evidence is overwhelming that Android is incompatible with the Java platform. *See, e.g.*, C.A. 2205, 21,181, 21,503-04.

Moreover, there is no split. App. 53-55 (collecting cases).[6] *Lotus* does not hold that code is not copyrightable simply because it allows compatibility. *Supra* 13. And, though *Lexmark* contains some dicta about compatibility, that was an observation about merger, and was not relevant to that court's dispositive originality holding. 387 F.3d at 536.

III. THE COURT OF APPEALS' DECISION IS CORRECT

The decision below is compelled by the Copyright Act's plain language and this Court's precedent. Google's reading of the statute amounts to a fundamental alteration of copyright law.

A. The Court Of Appeals' Interpretation Of § 102(b) Is The Only One Consistent With The Statutory Language And This Court's Decisions

1. The basic disagreement is over the relationship between 17 U.S.C. §§ 102(a) and 102(b). Section 102(a) grants copyright protection for "original works of authorship." Section 102(b) provides that "copyright protection for an original work of authorship" does not "extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery."

[6] Contrary to Google's characterization (Pet. 18), in assessing the copyrightability of plaintiffs' work, *Altai* considers constraints on plaintiffs in designing their original works, such as compatibility with already existing programs—not defendants' interest in compatibility with plaintiffs' work. 982 F.2d at 709-10.

The petition begins by misstating the court of appeals' interpretation. The court did not hold that "copyright protection extends to all elements of an original work of computer software, including a system or method of operation." Pet. i. It held that Oracle did not claim protection for a system or a method of operation but rather one of many ways of expressing a particular function or idea, such as establishing an internet connection. App. 47-48.

Nor did the court adopt "a vague 'greater scrutiny' test." Pet. 22. Rather, the court explained that §§ 102(a) and 102(b) codify the fundamental principle that copyright "protection extends only to the expression of an idea—not to the underlying idea itself." App. 18 (citing *Mazer*, 347 U.S. at 217). In other words, the court held that §§ 102(a) and 102(b) simply codify the age-old dichotomy between ideas and expression.[7]

This Court addressed the line between protected expression and unprotected ideas in *Baker v. Selden*, 101 U.S. 99 (1880). *Baker* involved a book describing a particular sort of dual-entry accounting system. This Court allowed the author to copyright the particularized expression of an accounting system he had devised: his "statement[s]" in the book describing his idea for accounting which "no one has a right to print or publish." *Id* at 104. But this Court held that the author could not copyright the very idea of an accounting system of that sort or the blank ac-

[7] The *Lexmark* dicta Google quotes (Pet. 15) agrees that § 102(b) "embodies the common-law idea-expression dichotomy." 387 F.3d at 534.

counting forms that "must necessarily be used as incident to" his accounting system. *Id.* at 104. The key distinction is whether protection was sought for an idea (unprotected) or expression of that idea (protected). *Id.* at 100. Accordingly, the defendant's work, which used "a different arrangement" and "different headings," permissibly used the idea for an accounting system but did not plagiarize the author's actual expression of the system. *Id.*

Applying those principles, the court of appeals held that Oracle sought no copyright for the "function[]" performed, such as "open[ing] an internet connection" or the concept of a "'package-class-method' structure." App. 46-48. What Oracle rightfully sought to protect is the concededly original "*particular* way of naming and organizing each of the 37 Java API packages," and just one of the "multiple ways" that the function can be described or expressed through Oracle's source code. *Id.* (emphasis in original).

In contrast, Google insists that § 102 does not codify the idea/expression dichotomy, but rather establishes a two-part analysis: A work described in § 102(a) is protected, subject to a list of "absolute" exclusions in § 102(b). Pet. 21. Under this reading, a work loses all copyright protection if it also describes "any idea, procedure, … concept, principle, or discovery." *Id.* § 102(b). Similarly, a work, such as computer code, would lose all copyright protection if it is also a "method of operation," "process," or "system." Under this logic, computer code could never be copyrightable. *Supra* 16.

Section 102's plain language supports the court of appeals' reading. The provision does not say that an original work loses copyright protection. It says that the author's rights cannot "extend" beyond the "original work," itself. 17 U.S.C. § 102(b). An author can secure copyright protection for an original work that also *describes* an accounting system but cannot "extend" his monopoly to preclude anyone else from using the accounting system just because he wrote about it. An author can secure copyright protection for an original literary work (e.g., the text of *West Side Story*) but that protection does not "extend" to the very "concept" or "idea" (of star-crossed lovers from feuding families).

This Court has resolved this issue. It held: "The idea/expression dichotomy is codified at 17 U.S.C. § 102(b)." *Golan v. Holder*, 132 S. Ct. 873, 890 (2012); *accord Eldred v. Ashcroft*, 537 U.S. 186, 219 (2003) (discussing § 102(b) in the context of the idea/expression dichotomy); *Harper & Row Publ'rs, Inc. v. Nation Enters.*, 471 U.S. 539, 556 (1985) (same). Indeed, the case on which Google most relies—*Feist*—rejects Google's reading explaining that "'Section 102(b) … restate[s] … that the basic dichotomy between expression and idea remains unchanged,'" 499 U.S. at 356 (quoting H.R. Rep. at 57; S. Rep. No. 473-94 at 54 (1975) *available at* 1975 WL 370212 ("S. Rep.")). In other words, copyright does not "extend" to a program's function, permitting the author to monopolize the function performed (like opening an internet connection), but does protect the original expression contained in the program's particular code and structure.

2. The legislative history this Court quoted in *Feist* applies that principle to computer code: "Section 102(b) … make[s] clear that the expression adopted by the programmer is the copyrightable element in a computer program" while "the actual processes or methods embodied in the program are not." H.R. Rep. at 57; S. Rep. at 54. Therefore, a "variety of ways to perform the same function sustains the classification of such works as 'expression'" of an idea—not the idea itself. *Atari Games Corp. v. Oman*, 888 F.2d 878, 885 (D.C. Cir. 1989) (Ginsburg, J.). Programmers do not own the "system" or "method of operation" because others can perform the same function using different code.

3. As we have seen, Google's reading of § 102(b) is especially problematic for computer code. *All computer code* is, as Google puts it, "a system or method of operating something," be it a computer or a computer program. Pet. 24. But Congress plainly intended to grant protection to "computer program[s]" when it defined them as "set[s] of *statements or instructions* to be *used directly or indirectly* in a computer in order *to bring about a certain result*." 17 U.S.C. § 101.

4. The court of appeals correctly applied § 102(b). As with merger (at 20, 22), the critical point is that "Oracle had unlimited options as to the selection and arrangement of the 7,000 lines [of code] Google copied." App. 33 (quotation marks omitted); *accord* App. 34, 48-49. Accordingly, Oracle does not seek to bar others from writing programs that function as Oracle's do but which use different code and different organization. Instead, Oracle seeks to protect on-

ly the "particular" expression it developed—the "particular" code and organization of its packages. App. 47.

Google dismisses the unlimited options available to Oracle's developers when they wrote and arranged the code. It asserts that "[i]f Google had not replicated the method headers exactly, code that used the shorthand commands based on those headers would not have run on Android." Pet. 30; *accord* Pet. 7, 31; *see* Pet. 29-30 (claiming Google copied so Java programmers would feel comfortable programming for Android). The court of appeals correctly concluded that whatever constraints *Google* might have felt when it was copying are irrelevant to the copyrightability of Oracle's work written a decade earlier. App. 33-34, 54-55. "Copyright in a work … *subsists from its creation* and … *endures* for [the copyright] term." 17 U.S.C. § 302(a); *accord* H.R. Rep. at 136 ("a work" is "protected by statutory copyright from its creation and … endures" (quotation marks omitted)); S. Rep. at 119 (same). Copyright "protect[s] all works of authorship *from the moment of their fixation in any tangible medium of expression*." National Commission on New Technological Uses of Copyrighted Works, Final Report 21 (1978).

5. Google's erroneous reading of § 102(b) finds no support in the supposed "fundamental boundary between patent and copyright law." Pet. 20. This Court rejects any such boundary: "Neither the Copyright Statute nor any other says that because a thing is patentable it may not be copyrighted." *Mazer*, 347 U.S. at 217. There is "nothing in the copyright statute to support the argument that the intended use or use in industry of an article eligible for copyright

bars or invalidates its registration. We do not read such a limitation into the copyright law." *Id.* at 218.

Google worries that the standard for copyrightability is too low. Here, again, Google is tilting at this Court's opinions. This Court holds that, in contrast to patent eligibility, the Copyright Act's threshold for copyright protection is "extremely low." *Feist*, 499 U.S. at 345. No "novelty" is required; any "creative spark" is sufficient, "no matter how crude [or] humble." *Id.* (quotation marks omitted).[8] That is not a problem, precisely because "there will nearly *always* be more than one way to write software code to accomplish a particular function." Pet. 26 (emphasis in original). Thus, the author of original code cannot stop others from writing code in any of those other ways. Copyright simply prohibits a plagiarist from copying the same code and organization.

Copyright's low threshold is also not a problem because, contrary to Google's assertion (Pet. 33-34), there are other defenses to infringement, such as fair use. The court below emphasized that certain arguments about "interoperability"—the focus of Google's

[8] The court of appeals rejected Google's "absolute" categorical exclusions, Pet. 21, in favor of a "nuanced assessment," App. 24-25, that considers whether the work "qualifies as an expression," App. 43. Thus, the decision below stands in stark contrast to Federal Circuit decisions Google cites (Pet. 27-29) that this Court has reviewed. *See, e.g., KSR Int'l Co. v. Teleflex, Inc.*, 550 U.S. 398, 415 (2007) ("rejecting the [Federal Circuit's] rigid approach" to obviousness in favor of an "expansive and flexible approach").

petition—"may be relevant to a fair use analysis," App. 68.

Equally ineffective is Google's QWERTY analogy. No case has ever addressed the copyrightability of QWERTY. But even assuming an "infringement lawsuit" on QWERTY would "fail[]," Pet. 2, that has no bearing here. There is no comparison between a work consisting of an arrangement of 26 unprotectable letters, which was not intended to be expressive, and a work consisting of 7000 unique lines of code.

Google suggests that QWERTY proves that one cannot copyright an original work that *also* performs a function. But this Court disagrees. *Mazer*, 347 U.S. at 218. If Google were right, there would be no copyright protection for the "West Key Number System, … the Bluebook[], … [b]lueprints …, instruction manuals … , used car value guides, dictionaries, encyclopedias, maps," etc. *Am. Dental Ass'n v. Delta Dental Plans Ass'n*, 126 F.3d 977, 978 (7th Cir. 1997) (Easterbrook, J.).

B. The Decision Below Promotes Innovation

The Founders, Congress, and this Court determined that "the best way to advance public welfare" is to "encourage[]" authors to engage in exactly the sort of "individual effort" Oracle undertook, by offering them "personal gain" in return. *Mazer*, 347 U.S. at 219; *accord* U.S. Const. art. I, § 8, cl. 8; 17 U.S.C. § 106; *accord Eldred,* 537 U.S. at 245; *Sony Corp. of Am. v. Universal City Studios, Inc.*, 464 U.S. 417, 429 (1984); *Twentieth Century Music Corp. v. Aiken*, 422 U.S. 151, 156 (1975). Oracle would never have "invested as heavily in Java" if it knew its "investment … would not receive copyright protection." McNealy C.A. Amicus Br. 3; *see* Microsoft C.A. Amicus Br. at 3-4, 8; Former Copyright Register Ralph Oman C.A. Amicus Br. 2-3 (denying copyright protection here "will chill investment and innovation in the software industry, and retard development of future generations of software").

Contrary to Google's suggestion that the decision below undermines the software industry's settled "understanding" of copyright law, Pet. 36, even Google's amici confirm that businesses "rely on that protection [for computer code] in their respective businesses," H.P. Br. 3; *see also* BSA C.A. Amicus Br. 31 ("[s]oftware companies … rely on … copyrightability of software"). Absent robust copyright protection, "software developers would be subject to substantial uncertainty, creating disincentives to develop new software and hampering innovation." BSA C.A. Amicus Br. 1-2; *id.* at 34 (Google's approach "undermine[s] the protections for innovation—and the corresponding incentives to innovate—that have served the public, the economy, and the software industry so well over the past 30 years").

Google is wrong that unauthorized copying of others' code is just how the software industry works. Pet. 33-36. All of Google's illustrations involved either no copying or a license. For example, Google's sources acknowledge that Compaq "re-implemented" IBM's BIOS System by writing "completely different code, skirting any copyright issues." Charles H. Ferguson & Charles R. Morris, *Computer Wars: The*

Post-IBM World 53 (1993) (cited by Pet. 33). In Google's examples regarding Apple and Eucalyptus (Pet. 33-34), both took licenses "to build on existing" platforms.[9] So even when "innovation depends on software developers' ability to build on what has come before," Pet. 1, it never depended on—and indeed would be defeated by—software developers' ability to misappropriate what came before.

The Java platform is the best example of how copyright protection promotes software innovation. The Java platform proliferated successfully, not because programmers were permitted to copy code without permission, but because Oracle created and licensed sophisticated, easy-to-use software that enabled programmers to "write once, run anywhere." *Supra* 4, 7-8. Google could have been part of Oracle's innovative ecosystem but rejected it and adopted an *in*compatible alternative. *Supra* 8.

It is, therefore, ironic that Google asserts that its approach best avoids lock-in. Pet. 30-32. Oracle, not Google, promoted "write once, run anywhere" so programmers would no longer have to write different versions of programs for different operating systems or devices. App. 4. But, because Google deliberately made Android incompatible, when a programmer

[9] *See* FreeBSD Project, *The FreeBSD Copyright* (2014), https://www.freebsd.org/copyright/freebsd-license.html; NetBSD Project, *NetBSD Licensing and Redistribution* (2014), http://www.netbsd.org/about/redistribution.html; *Amazon Web Services (AWS) And Eucalyptus Partner To Bring Additional Compatibility Between AWS And On-Premises IT Environments* (Mar. 22, 2012), https://www.eucalyptus.com/news/amazon-web-services-and-eucalyptus-partner.

writes for Android, that programmer and program are locked *into* Android and locked *out* of the Java platform. C.A. 21,179, 21,503-04.

Google and its amici raise alarms about the death of compatibility. Pet. 33-37. These arguments drastically over-read the opinion below to protect anything that might be called an "interface," H.P. Br. 6-9, 15-16; Public Knowledge Br. 5; Open Source Initiative Br. 10-11, 15; "format of data files," Pet. 35; H.P. Br. 9; or "interface specifications," CCIA Br. 3-4. That is false. The court of appeals merely held that thousands of lines of concededly original and highly creative source code are eligible for copyright protection. That does not mean that every work ever called an "API," "interface," "data file format," or "interface specification" is copyright protected. Many of them are not sufficiently original or expressive. Accordingly, the court of appeals' decision does not lead to the parade of horribles Google portrays.

If Google had been so concerned about compatibility and lock-in, it could have created its own compatible platform or taken a license. Google did neither. Pet. 7-8; App. 9-10.

IV. THIS INTERLOCUTORY APPEAL IS A POOR VEHICLE FOR THE QUESTION PRESENTED

A. The Judgment Below Is Supported By An Independent Rationale Not Challenged Here

Although one would never know it from Google's petition, the court of appeals found that Oracle's work is protected on two separate bases: (1) the original declaring code and (2) the packages' creative structure and organization. App. 3, 17, 77. Focused on declaring code, Google's petition does not address the structure-and-organization rationale. Google's lead argument, in fact, is inapplicable to structure and organization: Whatever might be said about declaring code, the structure and organization is not used "to operate the methods, *i.e.*, the pre-written programs." Pet. 6; *accord* Pet. 30.

Because Google's petition ignores an adequate alternative basis for affirmance, it must be denied. *See Goldlawr, Inc. v. Heiman*, 369 U.S. 463, 465 n.5 (1962); *see also supra* 18-24 (explaining the structure-and-organization rationale is correct and presents no cert-worthy issue).

B. Google Waived Its Only Argument Here

Worse yet, Google's only question presented—as it relates to code—is waived. Google asks whether "copyright protection extends to … computer software, including a system or method of operation." Pet. i. In this Court, Google contends that Oracle's declaring code is an uncopyrightable "system" or "method of operation." Pet. 29-32. But in the court of appeals, Google made no such argument. That court's decision carefully catalogs the arguments Google made as to each element. App. 28. As that list confirms, for declaring code, Google asserted only a waiver argument and a merger argument, Google C.A. Op. Br. 65-67—both of which Google abandons here.

Google asserted "method of operation," only as to the packages' structure and organization. *Id.* at 57-65. That is why the court of appeals addressed "method of operation" only as to structure and organization. App. 41-49. Similarly, the court of appeals (App. 30-37) reversed the two bases on which the district court denied protection to the declaring code: the short phrases doctrine and merger, App. 155-56. Yet, as evident from the question presented, Google's petition does not challenge either of the court of appeals' holdings on these two issues.

Google attempts to paper over its waiver by asserting that "[a]ll of Oracle's claims" "challenged the same thing: Google's replication of the method headers." Pet. 8. But the code is not "the same thing" as the packages' structure and organization—the "'groupings [of methods] among the various classes and packages,'" App. 49 (quoting App. 104)—which is why the courts below addressed them separately. Google cannot overcome its strategic decision to make certain arguments as to only one theory by now conflating them. Its argument is waived. *Kennedy v. Plan Adm'r for DuPont Sav. & Inv. Plan*, 555 U.S. 285, 290 n.2 (2009).

C. The Court Of Appeals' Decision Is Interlocutory

That this appeal is interlocutory renders it "not yet ripe for review." *Brotherhood of Locomotive Firemen v. Bangor & Aroostook R.R.*, 389 U.S. 327, 328 (1967) (per curiam). The court of appeals remanded for retrial on fair use, App. 58-70, which will resolve whether Google's copying is nevertheless "not an infringement of copyright," 17 U.S.C. § 107. Some courts have sanctioned narrow, limited copying—quite different from Google's copying—as a fair use. *See, e.g.*, *Sega Enters. Ltd. v. Accolade, Inc.*, 977 F.2d 1510, 1519 (9th Cir. 1993). And *Lotus*'s concurring judge described "fair use" as a "different approach" to reach the same copyright ends and vindicate the same interests Google and its amici raise. 49 F.3d at 821 (Boudin, J.). Using the QWERTY example, Judge Boudin suggested that perhaps "fair use" "might more closely tailor the limits on copyright protection to the reasons for limiting that protection." *Id.*

Unlike *Lotus*, fair use has not been decided. Pet. 29. But Google and its amici's heavy reliance on QWERTY and fair-use cases, like *Sega*, *see* Law Prof. Amicus Br. 6-7, 15-18, 21-22, illustrate that this Court could benefit from a fully developed fair-use record. *See* Pet. 27 (arguing the court below "transformed … copyright eligibility into … part of a fair-use defense").

Google suggests the software industry cannot wait for the remand. Pet. 33, 36-37. But, by Google's own account, the split it claims has existed for decades. Yet the phenomenal software growth Google celebrates erupted during that period, particularly in jurisdictions, such as the Ninth Circuit, that long ago adopted the court of appeal's approach here. The software industry will continue to thrive. And if Google is correct that this is "a Recurring Question," Pet. 32, the issue will recur before this case wends its way back.

For now, the interlocutory posture is "alone" "sufficient" to deny the petition. *Hamilton-Brown Shoe Co. v. Wolf Bros. & Co.*, 240 U.S. 251, 258 (1916).

CONCLUSION

This Court should deny the petition.

Respectfully submitted,

E. Joshua Rosenkranz
Counsel of Record
ORRICK, HERRINGTON &
SUTCLIFFE LLP
51 West 52nd Street
New York, NY 10019
(212) 506-5000
jrosenkranz@orrick.com

Date: December 8, 2014