

No. 18-956

IN THE
Supreme Court of the United States

GOOGLE LLC,

PETITIONER,

v.

ORACLE AMERICA, INC,

RESPONDENT.

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

**BRIEF OF *AMICI CURIAE* MOZILLA CORP.,
MAPBOX, A MEDIUM CORP., PATREON,
ETSY, INC. AND WIKIMEDIA FOUNDATION
IN SUPPORT OF PETITIONER**

JASON M. SCHULTZ
(*COUNSEL OF RECORD*)
NYU TECHNOLOGY LAW AND POLICY CLINIC
NYU SCHOOL OF LAW
245 SULLIVAN STREET, 609
NEW YORK, NY 10012
TELEPHONE: (212) 992-7365
JASON.SCHULTZ@LAW.NYU.EDU

Counsel for Amici Curiae

TABLE OF CONTENTS

INTERESTS OF *AMICI CURIAE* 1

SUMMARY OF ARGUMENT..... 3

ARGUMENT 5

 I. Introduction 5

 II. The Court should grant certiorari to rectify the Federal Circuit’s legal mistakes that threaten to chill innovation and inhibit competition in the software field. 9

 A. The Federal Circuit’s ruling upends decades of industry practice where software developers have relied on clear legal rules that allow for reimplementations of APIs without fear of copyright liability. 10

 B. The Federal Circuit’s decision increases barriers to entry in the software industry by reducing efficiency and increasing litigation risk. 11

 III. The Federal Circuit’s decision conflicts with several bedrock copyright precedents that software engineers have relied upon for decades. 15

A. The Federal Circuit failed to recognize that transformative uses in the software context can include “new opportunities” to reimplement API SSOs.	17
B. The Federal Circuit failed to follow this Court’s <i>Campbell</i> case which held that the amount and substantiality of the original work taken need only be “reasonable” in light of the purpose instead of “necessary.”	19
CONCLUSION	20

TABLE OF AUTHORITIES

Cases

<i>A.V. ex rel. Vanderhye v. iParadigms, LLC</i> , 562 F.3d 630 (4th Cir. 2009).....	18
<i>Authors Guild v. Google, Inc.</i> , 804 F.3d 202 (2d Cir. 2015).....	18
<i>Authors Guild, Inc. v. HathiTrust</i> , 755 F.3d 87 (2d Cir. 2014)	18
<i>Baker v. Selden</i> , 101 U.S. 99 (1879).....	3, 15, 16
<i>Campbell v. Acuff-Rose Music, Inc.</i> , 510 U.S. 569 (1994).....	4, 17, 19, 20
<i>Lotus Development Corp. v. Borland International, Inc.</i> , 49 F.3d 807 (1st Cir. 1995), <i>aff'd by an equally divided court</i> , 516 U.S. 233 (1996).....	15
<i>Oracle Am., Inc. v. Google LLC</i> , 750 F.3d 1339 (Fed. Cir. 2014)	16, 17
<i>Oracle Am., Inc. v. Google LLC</i> , 886 F.3d 1179 (Fed. Cir. 2018)	17, 19
<i>Perfect 10, Inc. v. Amazon.com, Inc.</i> , 508 F.3d 1146 (9th Cir. 2007).....	17
<i>Sega Enterprises Ltd. v. Accolade, Inc.</i> , 977 F.2d 1510 (9th Cir. 1992).....	4, 16, 17, 18
<i>Sony Computer Entertainment, Inc. v. Connectix Corp.</i> , 203 F.3d 596 (9th Cir. 2000).....	4, 16, 17, 18

Statutes

17 U.S.C. § 102(b).....	3, 15, 16
17 U.S.C. § 107(3).....	19

Other Authorities

- Browser Extensions*, MDN Web Docs,
[https://developer.mozilla.org/en-US/Add-
 ons/WebExtensions](https://developer.mozilla.org/en-US/Add-ons/WebExtensions) 10
- Byron Deeter, *The API-Economy Is Coming and
 Fast*, VentureBeat (Aug. 31, 2013, 2:00 PM) 8
- Chris Riley, *Mozilla Files FTC Comments
 Calling for Interoperability to Promote
 Competition*, Mozilla: Open Pol’y & Advoc.
 (Aug. 21, 2018) 15
- Letter from Chris Riley, Dir., Pub. Policy,
 Mozilla Corp., to Office of the Sec’y, Fed.
 Trade Comm’n (Aug. 20, 2018) 15
- Matt Murphy & Steve Sloan, *The Rise of APIs*,
 TechCrunch (May 21, 2016) 8
- Microsoft Edge Documentation: Extensions*,
 Microsoft Docs, [https://docs.microsoft.com/en-
 us/microsoft-edge/extensions](https://docs.microsoft.com/en-us/microsoft-edge/extensions) 10
- Microsoft Edge Documentation: Porting an
 Extension from Chrome to Microsoft Edge*,
 Microsoft Docs, [https://docs.microsoft.com/en-
 us/microsoft-edge/extensions/guides/porting-
 chrome-extensions](https://docs.microsoft.com/en-us/microsoft-edge/extensions/guides/porting-chrome-extensions) 10
- Venkat Atluri, Miklos Dietz & Nicolaus Henke,
*Competing in a World of Sectors Without
 Borders*, McKinsey Q. (McKinsey & Co., New
 York, NY), no. 3, 2017 8

INTERESTS OF *AMICI CURIAE*¹

Mozilla Corporation has been a pioneer and advocate for the web for more than a decade. Mozilla creates and promotes open standards that enable innovation and advance the web as a platform for all. Today, hundreds of millions of people worldwide use Mozilla Firefox to discover and experience the web on computers, tablets, and mobile phones.

Mapbox is a growing startup founded in Washington, D.C., with more than 500 million users interacting with its technology each month. Despite offering products that compete with Google Maps, Mapbox's interests in this case concern the bigger picture. Balance and predictability in copyright law are vital to innovation as a whole in the software industry. As a provider of online services, Mapbox is intimately familiar with APIs, providing many such interfaces to its customers. The possibility of copyright protection did not motivate Mapbox to make these interfaces; ease of use for customers did.

A Medium Corporation (“Medium”) provides an online publishing platform where people can read, write, and discuss the ideas of the day. Medium's ecosystem connects users with thoughtful, long-form writing by leaders, thinkers, entrepreneurs, artists,

¹ Parties' counsel were given timely notice of amici's intent to file this brief pursuant to the requirements of Rule 37.2(a). A copy of Petitioner's letter indicating consent has been filed with the Clerk of this Court. Respondent consented via email to Counsel of Record for Amici. No counsel for either party has had any role in authoring this brief, and no persons other than amici and their counsel have made any monetary contribution to the preparation or submission of this brief. See Rule 37.6.

and journalists. About 90 million people read on Medium each month.

Patreon is a membership platform making it easy for creators to get paid by their fans. Patreon has sent over \$500M to creators since its founding, which is made possible because of the many API-based integrations with its partners to allow creators to offer membership across the internet.

Etsy, Inc., and the two million creative entrepreneurs who sell on Etsy, rely on open standards to help make Etsy's marketplace flourish. Etsy's marketplace connects millions of buyers to sellers from nearly every country in the world for unique, handcrafted and vintage products.

Wikimedia Foundation is a non-profit organization based in San Francisco, California, which operates twelve free-knowledge projects on the Internet, including Wikipedia. Wikimedia's mission is to develop and maintain factual and educational content created and moderated by volunteer contributors, and to provide this content to people around the world free of charge. Additionally, the Foundation writes free and open source software to enable people worldwide to implement wiki-style information exchanges for their own usage. The MediaWiki software that the Foundation develops has been reimplemented by corporations, educational institutions, and government agencies to record and share information.

SUMMARY OF ARGUMENT

Competition and innovation are two principles at the heart of a healthy internet and the field of software development that fuels it. For decades, software engineers have relied heavily on reuse and reimplementations of functional protocols, such as the Application Programming Interfaces (APIs) in this case, to create competing alternatives to incumbent industry players and new markets for development without fear of copyright infringement. In accord with this Court's ruling in *Baker v. Selden*, 101 U.S. 99 (1879), and the plain language of 17 U.S.C. § 102(b) (2012), the software industry has flourished utilizing this approach to make internet and software ecosystems more accessible, affordable, diverse, and robust.

By reversing this rule in the context of APIs, the Federal Circuit upended decades of industry practice and the well-established expectations of developers, investors, and consumers. API reimplementations are a common theme among developers of all sizes—from those wishing to create entirely new platforms to those wishing to develop on them. The court below heedlessly unraveled this reasonably predictive rule and set of reliable norms that are critical to software coders for understanding what is appropriate to carry over from one project to another and what is not. This is especially true for individual coders, small startups, or nonprofit software projects, who often lack legal counsel or large financial reserves to defend themselves against unwarranted litigation.

Amici urge the Court to grant Google's petition for certiorari in order to correct this misreading of copyright law. Specifically, Amici wish to highlight two fundamental concerns with the lower court's opinion. First, the court's dramatic expansion of copyright protection to include APIs, which Amici believe are not copyrightable under U.S. law, stifles innovation and competition by privileging powerful incumbents and creating artificial barriers to entry for new players and innovators where none existed before. Second, the Federal Circuit's rejection of the fair use doctrine stands to undermine not only reimplementations and reuse of APIs, but also other valuable software engineering practices, such as reverse engineering, interoperability, and the creation of competing platforms, as well as innovations in data analytics, search engines, and many other groundbreaking advancements. Specifically, by creating irreconcilable conflicts with bedrock software fair use principles that have set the norms of engineering practice for over two decades, the Federal Circuit has opened the door to relitigating many status quo software engineering practices—practices that open source projects and small startups depend on every day to produce new platforms, programs, features, and interfaces. *See, e.g., Campbell v. Acuff-Rose Music, Inc.*, 510 U.S. 569 (1994); *Sony Comput. Entm't, Inc. v. Connectix Corp.*, 203 F.3d 596, 606–07 (9th Cir. 2000); *Sega Enters. Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992).

For these reasons, we urge the Court to grant Google's petition for certiorari.

ARGUMENT

I. Introduction

Application Programming Interfaces (APIs) serve countless functions in the software world. From helping the software running your phone to maintaining medical equipment to supporting everything you do on a computer, it would be impossible to list them all. At issue in this case is a particular set of APIs for mobile operating systems, but the implications of the ruling below are much larger and have the potential to completely restructure the way in which software production, competition, and innovation occur, especially on the internet.

To help illuminate the importance of this case, we urge the Court to imagine APIs as similar to the electronic billing or shipping forms you might see when you go shopping online. When you buy something on an e-commerce website, you are typically asked to enter information related to shipping and method of payment. While each and every e-commerce site has a slightly different style, almost every payment screen asks you to fill out a nearly identical structured form: name, address, credit card or bank information, billing address, shipping address, etc. Sites will display these fields in various shapes, sizes, fonts, and colors, but the structure, sequence and organization (SSO) of the information the site requests to process your order are almost ubiquitous and arguably orthodox in their presentation. Why is this so? Given there are myriad possible methods of exchanging shipping and

payment information with websites, isn't the blatant copying of this SSO from one site to another a case of massive copyright infringement?

The obvious answer is no. While each shopping site could attempt to come up with a totally new and different method of requesting payment and shipping data, common sense, technological standardization, and economic efficiency have driven the industry to adopt an almost ubiquitous SSO that every user understands and with which they can easily interact. Forcing every new website with a payment page to reinvent an SSO for payment and shipping forms would impose enormous unwarranted costs, redirecting software developers' resources into unproductive endeavors and wasting customers' time in having to navigate a new form for every single site they visit. Copyright protection for shipping and payment SSOs would either unreasonably reward those who happen to code them first or drive new market entrants to adopt inefficient idiosyncratic shipping and payment forms that could easily frustrate and confuse, and even potentially mislead them into entering the wrong information, not to mention additional havoc for any of the numerous third-party products that "auto-fill" customer information. Such a result would hardly be considered a victory for innovation, creativity, or competition.

The above example is nontrivial. Online technologies and marketplaces such as e-commerce have provided immense benefits to society, especially because they offer low barriers for new entrants and relatively low-cost and low-stress opportunities for

consumers. Yet imagine if every newly formed online retailer was forced to require users to enter the same information but in a different format with different naming conventions. Instead of entering “First Name, Last Name,” the user might be required to enter “Name as it appears on most recent 1040 tax form” or “Name that comes after your first and middle names.” The information is substantively the same, but the inconvenience, confusion, and incoherence could easily drive customers away from new sites and back to dominant vendors that already have their information—namely, e-commerce megasites—not because they have better products or prices, but because they happen to have published intuitive or familiar interfaces and forms for entering the information required for the transactions.

The above concerns apply equally to APIs. Much as developers and customers of e-commerce websites have come to expect standardized SSOs for shipping and payment forms, developers of applications for operating systems often expect and depend on standardized SSOs for programming, e.g. APIs, especially if they work for startups or other small innovators. These engineers have very little time or resources to modify or adapt their applications to every bespoke platform, especially when each platform might have hundreds or even thousands of relevant APIs. Instead, when API SSOs are consistent across platforms, application developers are able to quickly and efficiently improve or adapt original products to new marketplaces, providing consumers with new choices and more competition. This approach succeeds because APIs are considered the raw materials of software development, and

developers therefore have always relied on the ability to reuse them.

The Federal Circuit’s ruling threatens this convention, forcing each operating system or platform developer to adopt ill-suited alternative protocols for interaction. It relies on the same logic and would produce the same absurd outcome as the shipping and payment form examples we describe above, but with much more serious technological and economic implications.² Interoperability and reimplementations benefit every layer of the innovation economy. For developers and inventors, the ability to reimplement APIs dramatically expands the opportunities for cooperative ventures that leverage the creativity of individuals and startups who lack the resources of major technology companies. For investors, this means more

² Consultants at McKinsey have estimated that as much as \$1 trillion in total economic profit globally could be up for grabs through the creation of “digital ecosystems” that will give customers a seamless or complementary experience through their various apps and devices. Venkat Atluri, Miklos Dietz & Nicolaus Henke, *Competing in a World of Sectors Without Borders*, McKinsey Q. (McKinsey & Co., New York, NY), no. 3, 2017, at 32, 38–39. APIs play a vital role in enabling these ecosystems because they are the linkage between how these applications and devices can communicate with one another to create this experience. API-driven mobile apps could generate up to \$25 billion in global revenues per year. Byron Deeter, *The API-Economy Is Coming and Fast*, VentureBeat (Aug. 31, 2013, 2:00 PM), <https://venturebeat.com/2013/08/31/api-economy/>. API re-usage also allows companies not only to get to market faster, but to spend more time on developing “their core capabilities” and differentiating their functionality “at higher velocity.” Matt Murphy & Steve Sloan, *The Rise of APIs*, TechCrunch (May 21, 2016), <https://techcrunch.com/2016/05/21/the-rise-of-apis/>.

opportunities to fund small businesses and startups with high growth potential. And for consumers, this means more choices and better services tailored to their needs. For these reasons, Amici urge this Court to grant certiorari and reaffirm that copyright does not stand in the way of APIs being used in socially, technologically, and economically beneficial ways.

II. The Court should grant certiorari to rectify the Federal Circuit's legal mistakes that threaten to chill innovation and inhibit competition in the software field.

Amicus Mozilla is home to a community spanning thousands of developers who write code that interacts with APIs on a daily basis. In this field, innovation happens through constant iteration and rapid movement. The ethos of the industry can be seen through projects like the Firefox browser, where thousands of developers both inside and outside Mozilla contribute to innovation every day, or at startup companies where employees work almost nonstop to get their products to market and just as hard to deal with the constant cycle of troubleshooting crashes, patching problems, and releasing new updates.

In ruling that API SSOs were both copyrightable and immune from the fair use defense, the Federal Circuit issued an edict that is both nonsensical and counterintuitive for most software developer communities. Expanding copyright to include API SSOs does not provide any incentives to create. Indeed, it puts up barriers to creation. Much like the

shipping and payment scenario we describe above, it results in confusion, wasted effort, and concern over exactly how and when interacting with another entity's API will lead to legal trouble.

A. The Federal Circuit's ruling upends decades of industry practice where software developers have relied on clear legal rules that allow for reimplementing of APIs without fear of copyright liability.

As noted above, much of the internet's compatibility and interoperability is based on the ease with which platforms, browsers, and other ubiquitous technologies can reimplement the functionality of core technologies. For example, Mozilla has adopted Google's "Extensions" API from the Chrome web browser to the Firefox browser.³ Mozilla's choice to support the Extensions API allows developers to build one extension and, after a few tweaks, deploy the extension in a number of different browsers, such as Google's Chrome browser, Mozilla's Firefox browser, and Microsoft's Edge browser.⁴ This increases the number of potential extensions available to all users, allowing them to

³ *Browser Extensions*, MDN Web Docs, <https://developer.mozilla.org/en-US/Add-ons/WebExtensions> (last visited Feb. 19, 2019).

⁴ *Microsoft Edge Documentation: Extensions*, Microsoft Docs, <https://docs.microsoft.com/en-us/microsoft-edge/extensions> (last visited Feb. 19, 2019); *Microsoft Edge Documentation: Porting an Extension from Chrome to Microsoft Edge*, Microsoft Docs, <https://docs.microsoft.com/en-us/microsoft-edge/extensions/guides/porting-chrome-extensions> (last visited Feb. 19, 2019).

easily enhance and add new functionality to their chosen browser, or if they wish, switch browsers as often as they like without high transaction costs.

Web development already works this way. With minor tweaks to the HTML code used to write webpages, the same pages will run in multiple browsers according to uniform protocols. Add-on development should be no different: With minor tweaks to the code, the same add-ons should run in multiple browsers according to uniform protocols, which in this case is the Extensions API SSO.

Mozilla's and Microsoft's reimplementations of the Extensions API is a classic example of software engineering norms. It was successful, in part, because there was no question that the API SSO could be adopted without restriction. This is not only what software engineers expect, but what makes sense for the success of the field as a whole with the obvious benefits to competition and innovation in the market for web browsing.

B. The Federal Circuit's decision increases barriers to entry in the software industry by reducing efficiency and increasing litigation risk.

While this case pits two technology giants against each other, Amici urge the Court to look beyond their size and valuation and consider the importance of the issues presented to smaller players in the software industry—including startups, individual developers, nonprofit projects, and consumer-innovators. In particular, we urge the

Court to consider two fundamental features of the software industry that have always enabled innovators of all sizes and means to contribute meaningfully to software development: (1) relatively clear copyright rules for product mapping, design, and development and (2) cheap access to the raw materials of software engineering, such as APIs, to build new products and ensure their usability in the software ecosystem.

Both features speak to the economics of startups and non-profit or community software projects. These efforts often involve the quintessential “garage” inventors—a few individual coders huddled together in a small office or home or remotely, working almost exclusively on personal computers to code as fast as they can to launch a new idea, product, or service into the world before their funds run out. In order to do this, copyright rules need to be relatively clear. For example, software engineers generally understand they cannot copy anyone else’s application source code unless it is under an “open source” license. This works as a practical rule, since often the purpose of the startup or project is to produce its own application, with its own code. However, in order to offer the new application to consumers on a range of existing platforms, operating systems, or browsers quickly and efficiently, copying and reimplementing APIs is commonplace, with the understanding that copyright does not and should not apply to these functional connectors. This allows for startups and small innovators to “plug-and-play” their applications across all technology ecosystems without suddenly having to negotiate copyright licenses for potentially

hundreds of APIs and/or rewrite the mechanism for their application to communicate with dozens or hundreds of alternative APIs to produce identical functionality. Under the Federal Circuit's approach, many startups, individual coders, or nonprofit software projects would have to shoulder these additional engineering, financial, and legal costs when extending the accessibility of their program to a new platform or operating system. Such additional burdens are excessive and unnecessary and would only reduce competition and innovation in the software field, not enhance them.

Oracle responds to these concerns by suggesting that SSO copyrights for APIs will simply force platform or operating system developers to create new API SSOs. But requiring awkward workarounds is "innovation" that the software engineers and the marketplace neither need nor desire. Moreover, Oracle fails to appreciate the added barriers this would impose on new entrants to technological ecosystems. Convincing application developers to rewrite their code for hundreds of new APIs every time they want to add it to a new platform is not only burdensome and expensive, but risky, as it may create new errors or incompatibilities that will require extensive quality assurance and maintenance. As many in the software industry would say, this simply does not scale; it is the equivalent of forcing every new e-commerce site to reinvent shipping and payment forms on their checkout webpage. While large companies may have enough capital to overcome these burdens and maintain their customer base, startups and smaller developers may find their

projects languishing—even if their technical approach is superior. The Federal Circuit’s decision sets up this inefficient, anticompetitive, and wasteful approach to software engineering.

By undermining the relative clarity of copyright rules for APIs, the Federal Circuit also created new litigation risks that will disproportionately impact startups, and smaller innovators who—unlike Google and Oracle—lack the resources to easily defend themselves and their software engineers. This could deter developers from creating new products that reimplement existing APIs, for fear their ideas will run into legal obstacles. And for individual coders, small startups, or nonprofit software projects, who lack legal counsel and large financial reserves, a simple cease-and-desist letter from a large software company could effectively shut down the launch of a new product or service that relies on that company’s API SSO.

On a broader level, the Federal Circuit’s ruling threatens competition across the entire software industry. The court’s dramatic expansion of copyright doctrine to annex the functional aspects of APIs will stifle innovation and competition by privileging powerful incumbents and creating a lock-out effect for new products. This would lead to an overall decrease in choice, both for innovators and consumers. Amicus Mozilla has long advocated for technical interoperability as essential to preserving consumer choice and economic competition on many

fronts.⁵ Without this Court's intervention, the Federal Circuit's ruling has the potential to reinforce the dominance of industry giants by increasing their proprietary leverage over small developers and other new entrants.

III. The Federal Circuit's decision conflicts with several bedrock copyright precedents that software engineers have relied upon for decades.

While ostensibly limited to the legal status of API SSOs, the Federal Circuit's decisions on copyrightability and fair use conflict with several of the bedrock copyright precedents that software engineers rely upon every day. In addition to this Court's opinion in *Baker v. Selden*, 101 U.S. 99 (1879), established into law by Congress in 17 U.S.C. § 102(b), and reaffirmed by courts in the software context in *Lotus Development Corp. v. Borland International, Inc.*, 49 F.3d 807 (1st Cir. 1995), *aff'd*

⁵ As Amicus Mozilla argued in a recent blog post, accompanying its FTC comment on the topic of competition in the Internet sector, "If the future of the internet stays grounded in standards and built out through an ecosystem of transparent third-party accessible APIs, we can preserve the digital platform economy as a springboard for our collective social and economic welfare, rather than watching it evolve into an oligarchy of gatekeepers over our data." Chris Riley, *Mozilla Files FTC Comments Calling for Interoperability to Promote Competition*, Mozilla: Open Pol'y & Advoc. (Aug. 21, 2018), <https://blog.mozilla.org/netpolicy/2018/08/21/mozilla-files-ftc-comments-calling-for-interoperability-to-promote-competition/>; see also Letter from Chris Riley, Dir., Pub. Policy, Mozilla Corp., to Office of the Sec'y, Fed. Trade Comm'n (Aug. 20, 2018), <https://blog.mozilla.org/netpolicy/files/2018/08/Mozilla-FTC-filing-8-20-2018.pdf>.

by an equally divided court, 516 U.S. 233 (1996), and *Sega Enterprises Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992), the decision below blurred and potentially undermines numerous software fair use holdings. While Amici do not believe that these holdings are in fact altered, we are concerned about the chilling effect this new uncertainty could have on innovation.

For example, the Federal Circuit’s decision on copyrightability appeared to read an exception into § 102(b)’s “method of operation” for “expression embodied in a method of operation,” and concluded that APIs fell under this exception. *Oracle Am., Inc. v. Google LLC*, 750 F.3d 1339, 1356–57 (Fed. Cir. 2014). Yet because this logic would apply beyond API SSOs to almost every single aspect of software engineering that involves computer code, it will be much more challenging to ascertain which of the myriad engineering practices employed across hundreds of thousands of software projects to utilize functional code arguably contain expression embodied in a method of operation. Determining which aspects fall under the Federal Circuit’s exception and which do not is left wide open by the opinion below, jeopardizing the relative clarity that *Baker* and § 102(b) were meant to provide.

The Federal Circuit also failed to consider the bedrock fair use holdings in *Sega* and *Sony Computer Entertainment, Inc. v. Connectix Corp.*, 203 F.3d 596 (9th Cir. 2000), mistakenly holding that software interoperability (performing an identical function) served as a “substitution” which is disfavored under the first factor of the fair use doctrine instead of a

“transformation” which is favored. *Oracle Am., Inc. v. Google LLC*, 886 F.3d 1179, 1210 (Fed. Cir. 2018) (holding that Google’s use of the API SSOs and accompanying declaring code “for the same purpose” as Oracle seriously weakened the fair use claim). *But see Sony*, 203 F.3d at 606 (finding Connectix’s emulation software to be modestly transformative despite similar uses and function to Sony’s because it “affords new opportunities for game play in new environments”); *Sega*, 977 F.2d at 1522 (holding that Accolade’s “ultimate purpose” in copying was to create compatibility between its videogames and Sega’s video game console). Nothing in the opinion below meaningfully clarifies the difference between copying code for the same purpose, copying code for compatibility, and copying code for similar uses and functions to afford new opportunities. This Court should grant certiorari to rectify these inconsistencies and return software fair use law to its previous stable status.

A. The Federal Circuit failed to recognize that transformative uses in the software context can include “new opportunities” to reimplement API SSOs.

As even the Federal Circuit recognized, a secondary use of computer code is more likely to be considered fair when it “changes” the underlying copyrighted work or uses it “in a different context” so that the work is “transformed into a new creation.” *Oracle Am., Inc. v. Google LLC*, 750 F.3d 1339, 1374 (Fed. Cir. 2014) (quoting *Perfect 10, Inc. v. Amazon.com, Inc.*, 508 F.3d 1146, 1165 (9th Cir. 2007)); *see also Campbell v. Acuff-Rose Music, Inc.*,

510 U.S. 569, 579 (1994) (“[T]he more transformative the new work, the less will be the significance of other factors . . . that may weigh against a finding of fair use.”). However, the court then misinterpreted this rule to limit its application strictly to situations where new code emerges from the secondary use. Such a limited and narrow ruling fails to capture the full range of transformative or “new” uses that occur in software, especially through engineering processes such as reimplementation, or when the API SSO is reused to perform a novel function or expand that functionality onto a new platform. *Sony*, 203 F.3d at 606–07 (finding that simply introducing existing computer code into a new context can be transformative within the practice of software engineering); *Sega*, 977 F.2d at 1522; *see also* *Authors Guild v. Google, Inc.*, 804 F.3d 202 (2d Cir. 2015); *Authors Guild, Inc. v. HathiTrust*, 755 F.3d 87 (2d Cir. 2014); *A.V. ex rel. Vanderhye v. iParadigms, LLC*, 562 F.3d 630 (4th Cir. 2009).

Such uses are a major form of innovation in the software industry and would be imperiled if the Federal Circuit’s ruling is left standing. The relevant question in a fair use analysis of reimplemented API SSOs should therefore not be whether the use transforms the SSO itself into a new structure, but whether the implementation as a whole—including the API SSO—is a new and different work. A programmer reusing an existing piece of code can radically depart from the underlying context of the original work by implementing the package differently while retaining the basic SSO to ensure the work remains compatible with others. These departures can be transformative, “add[ing]

something new, with a further purpose or different character” to the SSO. *Campbell*, 510 U.S. at 579. Developers frequently innovate by introducing creative ideas into new contexts, or expanding the user and developer bases for languages and functional systems.

B. The Federal Circuit failed to follow this Court’s *Campbell* case which held that the amount and substantiality of the original work taken need only be “reasonable” in light of the purpose instead of “necessary.”

In addressing the third fair use factor, the Federal Circuit committed legal error in requiring secondary users of API SSOs to show their use was “necessary” in light of their purpose instead of merely “reasonable” as this Court held to be the test in *Campbell*. Compare *Oracle Am., Inc. v. Google LLC*, 886 F.3d 1179, 1205–06 (Fed. Cir. 2018), with *Campbell*, 510 U.S. at 586 (“The third factor asks whether ‘the amount and substantiality of the portion used in relation to the copyrighted work as a whole’ . . . are reasonable in relation to the purpose of the copying.” (citation omitted) (quoting 17 U.S.C. § 107(3))). This distinction is critical for the viability of the software development field. Copying practices that are reasonable, such as reimplementing API SSOs, are relatively easy for programmers to develop and follow over time. They encourage industry norms and best practices. On the other hand, requiring software engineers to prove absolute necessity before they may copy even the most basic functional SSOs would impose huge additional costs, inefficiencies, and barriers to entry, especially for small and new

entrants. *Campbell's* “reasonable” test has been in place for almost 25 years and has served the software development community well; this Court should not allow the Federal Circuit to narrow it arbitrarily.

CONCLUSION

For the foregoing reasons, Amici ask this Court to grant certiorari in this case.

Respectfully submitted,

JASON SCHULTZ
Counsel of Record
NYU TECHNOLOGY LAW AND POLICY CLINIC
NYU SCHOOL OF LAW
245 Sullivan Street, 609
New York, NY 10012
Telephone: (212) 992-7365
Jason.schultz@exchange.law.nyu.edu

Counsel for Amici Curiae