Nos. 17-1118, -1202

IN THE UNITED STATES COURT OF APPEALS
FOR THE FEDERAL CIRCUIT

ORACLE AMERICA, INC.,

Plaintiff-Appellant,

v.

GOOGLE INC.,

Defendant-Cross-Appellant.

On Appeal from the United States District Court
for the Northern District of California
No. 3:10-cv-03561-WHA
Hon. William H. Alsup

BRIEF OF AMICI CURIAE ENGINE ADVOCACY, THE APP DEVELOPERS ALLIANCE, AND GITHUB INC.

Mark A. Lemley
Clement S. Roberts
Joseph C. Gratz
DURIE TANGRI LLP
217 Leidesdorff Street
San Francisco, CA 94111
(415) 362-6666
mlemley@durietangri.com

Counsel for Amici Curiae
Engine Advocacy, The App Developers Alliance, and GitHub, Inc.

CORPORATE DISCLOSURE STATEMENT

Pursuant to Federal Rule of Appellate Procedure 26.1, the undersigned states that no publicly held corporation owns 10% or more of the stock of any of amici.

DATED: May 30, 2017

/s/ Mark A. Lemley
Mark A. Lemley
DURIE TANGRI LLP
Counsel for Amici Curiae Engine Advocacy, The App Developers Alliance, and GitHub, Inc.

CERTIFICATE OF COMPLIANCE WITH RULE 29(C)(4)(E)

Counsel for the parties did not author this brief in whole or in part. The parties have not contributed money intended to fund preparing or submitting the brief. No person other than Amici Curiae or their counsel contributed money to fund preparation or submission of this brief.

DATED: May 30, 2017

/s/ Mark A. Lemley
Mark A. Lemley
DURIE TANGRI LLP
Counsel for Amici Curiae Engine Advocacy, The App Developers Alliance, and GitHub, Inc.

TABLE OF CONTENTS

PAGE NO.

TABLE OF AUTHORITIES

PAGE NO.

Cases

Statutes

Other Authorities

IDENTITY AND INTEREST OF THE AMICI CURIAE

Engine Advocacy is a policy, advocacy, and research organization supporting startups as an engine for economic growth.

The App Developers Alliance is the world's leading advocate for software developers and the companies invested in their success. Alliance members include industry leaders in consumer, enterprise, industrial, and emerging software, and a global network of more than 75,000 developers.

GitHub, Inc. is a web-based social coding platform that enables communities of users to collaboratively develop open-source software projects. GitHub hosts over 58 million projects and welcomes more than 21 million monthly visitors. GitHub-hosted software projects may be applications designed for web or mobile devices, or may be the source code that powers entire businesses. Developers on GitHub work together, sharing code and knowledge. As such, GitHub has an interest in reducing barriers to collaboration and promoting innovation in software development.

Amici have no direct financial interest in the outcome of this case, but have a strong interest in seeing that the law continues to support rather than impede innovation in the software industry.

CONSENT OF THE PARTIES

Both parties have been notified of our intention to file this brief and have consented to its filing.

SUMMARY OF ARGUMENT

Copyright exists to promote creativity. As a result, the copyright laws strike a series of delicate balances in order to protect different groups of creators. Too much protection for one group thwarts the creative innovation of others. Achieving the right balance is particularly important when it comes to software because the interactive nature of software makes the risks of overprotecting existing software particularly great.

The Ninth Circuit, like others, has emphasized the importance of interoperability in computer software copyright cases. It has repeatedly held that parties are free to copy the elements of a computer interface necessary to write new and different programs that work with the plaintiff's existing program.

Software companies, and startups in particular, rely on that settled legal doctrine to build new and innovative products. Without it, developers would be at the mercy of proprietary platforms written in specific, rapidly obsolete computer languages and without the ability to create new and innovative products that are broadly accessible to consumers. The result of such a balkanized regime would be significantly less creativity—the very opposite of what copyright law is designed to achieve.

ARGUMENT

I. SETTLED NINTH CIRCUIT LAW PROTECTS INTEROPERABILITY IN SITUATIONS LIKE THIS

The Ninth Circuit has repeatedly interpreted the Copyright Act's fair use doctrine to protect the right of third parties to copy application program interfaces ("APIs") when necessary to make their products work with products made by the copyright owner or others. That is true even when the use of the API requires copying the computer code itself, not just the higher-level functional aspects of the API. And it is true even if the Defendant copies the API in order to compete directly with the plaintiff by producing a compatible system.

In *Sega Enterprises Ltd. v. Accolade, Inc*., 977 F.2d 1510 (9th Cir. 1992), *as amended* (Jan. 6. 1993), for example, Accolade wanted to make video games compatible with Sega's game console over Sega's objection. To make its games run on Sega's platform, Accolade copied the entirety of Sega's computer code in order to "reverse engineer" the code and extract only the APIs—the portions necessary to ensure compatibility. The Ninth Circuit held that to be a fair use even though it involved copying of the entirety of the code because making that copy was necessary to get access to the interface components—which the Ninth Circuit found to be "unprotectable." The Court emphasized that "because Accolade has a legitimate interest in gaining such access (in order to determine how to make its cartridges compatible with the Genesis console)" its copying of the code to replicate the interface components was a fair use. *Id*. at 1520.

The fact that Accolade sought to write its own original programs, not to copy Sega's programs, loomed large in the Ninth Circuit's analysis:

> Accolade copied Sega's software solely in order to discover the functional requirements for compatibility with the Genesis console—aspects of Sega's programs that are not protected by copyright. With respect to the video game programs contained in Accolade's game cartridges, there is no evidence in the record that Accolade sought to avoid performing its own creative work. Indeed, most of the games that Accolade released for use with the Genesis console were originally developed for other hardware systems. . . . [A]lthough Accolade's ultimate purpose was the release of Genesis-compatible games for sale, its direct purpose in copying Sega's code, and thus its direct use of the copyrighted material, was simply to study the functional requirements for Genesis compatibility so that it could modify existing games and make them usable with the Genesis console. . . . On these facts, we conclude that Accolade copied Sega's code for a legitimate, essentially non-exploitative purpose. . . .

Id. at 1522-23 (citations omitted).

Nor was the court troubled that Accolade engaged in verbatim copying of some program interfaces in order to achieve that legitimate compatibility purpose:

> [C]omputer programs are, in essence, utilitarian articles—articles that accomplish tasks. As such, they contain many logical, structural, and visual display elements that are dictated by the function to be performed, by considerations of efficiency, or by external factors such as compatibility requirements and industry demands. . . . When specific instructions, even though previously copyrighted, are the only and essential means of accomplishing a given task, their later use by another will not amount to infringement.

Id. at 1524 (internal quotation marks omitted) (citations omitted).

In *Sony Computer Entertainment, Inc. v. Connectix Corp.*, 203 F.3d 596 (9th Cir. 2000), the Ninth Circuit went further, holding that it was fair use to create an emulator of the Sony game console—copying the code not just to reverse engineer it but to test how programs worked with it—because the purpose was to produce a new product that worked with the old system:

> We find that Connectix's Virtual Game Station is modestly transformative. The product creates a new platform, the personal computer, on which consumers can play games designed for the Sony PlayStation. This innovation affords opportunities for game play in new environments, specifically anywhere a Sony PlayStation console and television are not available, but a computer with a CD-ROM drive is. More important, the Virtual Game Station itself is a wholly new product, notwithstanding the similarity of uses and functions between the Sony PlayStation and the Virtual Game Station.

Id. at 606. For the same reason, the fact that Sony might lose sales to the Connectix system did not militate against fair use on the fourth factor. "[B]ecause the Virtual Game Station is transformative, and does not merely supplant the PlayStation console, the Virtual Game Station is a legitimate competitor in the market for platforms on which Sony and Sony-licensed games can be played[,]" so the loss of market share was not attributable to copyright infringement, but to legitimate competition. *Id*. at 607.

Judged against this Ninth Circuit precedent, Google has a strong claim to fair use. Google did not simply copy Java. Instead, it took only what was necessary to make Java work on the Android phone system. That was an innovation. Java was designed for desktop computers and wasn't suited to use on a phone. Appx51938:6-19. Neither Sun nor Oracle succeeded in creating a smartphone operating system using Java. Appx50559-50560 at 560:17-561:4, Appx51896:12-17. Google created the first ever open-source mobile platform, Android. The Android platform was "revolutionary" and "completely different from any other approach." Appx50346-50347 at 347:14-348:7, Appx50347-50348 at 348:21-349:1. That new platform is overwhelmingly Google's work, not Sun's

or Oracle's. Android includes 15 million lines of code, Appx51247:4-7, only a tiny fraction of which include the Java APIs. Nor did Android even copy all of the APIs. Rather, it used declarations from only 37 of the 166 Java API packages, and Google wrote its own code to implement those specifications. Appx51098-51099 at 1097:19-1098:11. Those 37 APIs are, by Oracle's own admission, "not separable" from the Java programming language and are "fundamental" to implementing Java. Appx51014. So Oracle's argument that implementing those APIs is not fair use as a matter of law is tantamount to arguing that interoperability is not fair use as a matter of law.

Sega and *Sony* hold that a company is free to copy the entirety of a computer program in order to build a compatible product, even where that compatible product copies API code directly in the final product and even where that final product competes directly with the plaintiff. Google's use of selected Java APIs here, which did not involve copying of the code in the final product and did not involve a competing product at all, seems by comparison easy to justify as fair use under the Ninth Circuit's case law promoting interoperability.

II. INTEROPERABILITY PROMOTES INNOVATION

The legality of copying APIs and other interface components is well-established, and has been for a quarter-century. That is true not only in the Ninth Circuit but in all other circuits to have considered the issue. *See, e.g., Computer Assocs. Int'l, Inc. v. Altai, Inc.*, 982 F.2d 693, 710-15 (2d Cir. 1992); *Bateman v. Mnemonics, Inc.*, 79 F.3d 1532, 1539, 1543-45 (11th Cir. 1996); *Lexmark Int'l, Inc. v. Static Control Components, Inc.*, 387 F.3d 522, 542 (6th Cir. 2004), *reh'g*

en banc denied (2005); *Lotus Dev. Corp. v. Borland Int'l, Inc*., 49 F.3d 807, 815-19 (1st Cir. 1995), *aff'd*, 516 U.S. 233 (1996); *see also id*. at 821 (Boudin, J., concurring); *Assessment Techs. of WI, LLC v. WIREdata, Inc*., 350 F.3d 640, 644-45 (7th Cir. 2003); *Mitel, Inc. v. Iqtel, Inc*., 124 F.3d 1366, 1374-76 (10th Cir. 1997); Pamela Samuelson & Suzanne Scotchmer, *The Law and Economics of Reverse Engineering*, 111 Yale L.J. 1575, 1621-26 (2002). Congress weighed in to endorse this settled law when it enacted the Digital Millennium Copyright Act. While that Act made it illegal to circumvent a technical protection measure that controlled access to a copyrighted work, Congress was careful to exclude from that new law circumventing such a measure "for the sole purpose of identifying and analyzing those elements of the program that are necessary to achieve interoperability of an independently created computer program with other programs, ..." 17 U.S.C. § 1201(f)(1).

Computer programmers and software companies, including Amici, rely on that settled law. Computers and the Internet work because different programs and devices can communicate with each other. Interoperability makes that possible. Interoperability is the reason you can read a web site regardless of what Internet browser you use. Interoperability is the reason you can read documents on a PC even though someone wrote them on a Mac. Interoperability is the reason messages can pass from phone to computer to tablet and back again.

Interoperability is also critical to the development of the new Internet of Things ("IoT") that connects a wide array of devices beyond computers. IoT by definition depends on autonomous communication amongst wide range of devices.

That cannot happen without interoperable standards in the IoT market. Fragmentation of the market due to competing, proprietary standards will severely curtail the value of IoT as a whole. Considering VCs invested more than $1 billion in IoT startups in 2016,[1] interoperability in the IoT sector has substantial economic consequences.

Interoperability is particularly important to startups. Companies that develop apps for mobile phones are often small. They may not have the ability to write several different versions of a program from scratch, one for each hardware platform or incompatible programming language—much less to separately negotiate agreements with each such platform provider in the economy. By allowing an app developer to reach the widest possible market, legal protection for interoperability increases the number of creative new works produced each year. It also ensures that no one company, no matter how dominant its platform, gets to decide what web pages you can access, what files you can share, or what programs you can download.

Without the security that they would be able to investigate and use APIs, software developers would be at the mercy of platform and programming giants who could decide whether, when, and how anyone could write or use a computer program that ran on their system. Startups will not invest in new products—whether it be for mobile phones or video games or the Internet of Things—without

[1] Mikey Tom, *IoT Breakdown: VCs betting billions on the connected world*, PitchBook (Dec. 7, 2016), https://pitchbook.com/news/articles/iot-breakdown-vcs-betting-billions-on-the-connected-world.

confidence that their products will work on the dominant platforms. That is why the risk of overprotecting copyright is so much greater in software than in other areas. Giving too much protection to a song may incrementally discourage the creation of somewhat similar songs. Giving copyright owners control over interoperability risks shutting down the software development ecosystem altogether.

Oracle and its predecessor Sun are well aware of the benefits of interoperability. In a brief filed by the American Committee for Interoperable Systems, a trade association that claimed both Sun and Oracle as members in the 1990s, both companies argued that API copyrights should not be used to prevent the creation of interoperable programs. They wrote: "If the developer of one part of the environment can use copyright law to prevent other developers from writing programs that conform to the system of rules governing interaction with the environment—interface specifications, in computer parlance—the first developer could gain a patent-like monopoly over the system without ever subjecting it to the rigorous scrutiny of a patent examination." Brief Amicus Curiae of ACIS and CCIA in Support of Respondent, *Lotus Dev. Corp. v. Borland Int'l, Inc*., No. 94-2003, 1995 WL 728487, at **4-5 (U.S. Dec. 8, 1995).

Oracle had it right in that brief. Interoperability is important to the continued development of creative computer programs and to the health of the software industry. Oracle's effort to prevent interoperability in this case is particularly ironic because Java was itself developed as a way of creating interoperability across platforms. Mark A. Lemley & David McGowan, *Could*

Java Change Everything? The Competitive Propriety of a Proprietary Standard, 43 Antitrust. Bull. 715 (1998), https://papers.ssrn.com/sol3/papers.cfm?abstract_id=69148.

The Java slogan was "Write Once, Run Anywhere." Ninth Circuit copyright law allows software developers to do just that—write a single program that works on multiple platforms. That in turn encourages more creative work. As the Ninth Circuit explained in *Sega*, "Accolade's identification of the functional requirements for Genesis compatibility has led to an increase in the number of independently designed video game programs offered for use with the Genesis console. It is precisely this growth in creative expression, based on the dissemination of other creative works and the unprotected ideas contained in those works, that the Copyright Act was intended to promote." *Sega*, 977 F.2d at 1523. To use copyright, a law designed to promote creativity, to stifle it instead would be perverse.

CONCLUSION

For the foregoing reasons, the decision of the jury and of the District Court should be affirmed.

DATED: May 30, 2017

Respectfully submitted,

/s/ Mark A. Lemley

Mark A. Lemley
Clement S. Roberts
Joseph C. Gratz
DURIE TANGRI LLP

217 Leidesdorff Street
San Francisco, CA 94111
(415) 362-6666
mlemley@durietangri.com

Counsel for Amici Curiae Engine Advocacy, The App Developers Alliance, and GitHub, Inc.

CERTIFICATE OF COMPLIANCE

I hereby certify that pursuant to Fed. R. App. P. 32(a)(7)(C) and Ninth Circuit Rule 32-1 this brief complies with the typeface requirements of Fed. R. App. P. 32(a)(5) and (6), because it is written in 14-pt Times New Roman, and with the type-volume limitations of Fed. R. App. P. 29(d) and Ninth Circuit Rule 29-2(c), because it contains 4,227 words, excluding the portions excluded under Fed. R. App. P. 32(a)(7)(B)(iii). This count is based on the word count feature of Microsoft Word.

DATED: May 30, 2017

/s/ Mark A. Lemley
Mark A. Lemley
DURIE TANGRI LLP
Counsel for Amici Curiae Engine Advocacy, The App Developers Alliance, and GitHub, Inc.

CERTIFICATE OF SERVICE

I hereby certify that I electronically filed the foregoing with the Clerk of the Court for the United States Court of Appeals for the Federal Circuit by using the appellate CM/ECF system on May 30, 2017.

Participants in the case who are registered CM/ECF users will be served by the appellate CM/ECF system.

DATED: May 30, 2017

/s/ Mark A. Lemley
Mark A. Lemley
DURIE TANGRI LLP
Counsel for Amici Curiae Engine Advocacy, The App Developers Alliance, and GitHub, Inc.