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 *AMICUS CURIAE* OF THE
COMPUTER & COMMUNICATIONS INDUSTRY
ASSOCIATION IN SUPPORT OF PETITIONER**

————————◆————————

MATT SCHRUERS
ALI STERNBURG
COMPUTER & COMMUNICATIONS
  INDUSTRY ASSOCIATION
900 17th Street NW, Suite 1100
Washington, D.C. 20006
(202) 783-0070
mschruers@ccianet.org

JONATHAN BAND
  Counsel of Record
JONATHAN BAND PLLC
21 Dupont Circle NW,
  Suite 800
Washington, D.C. 20036
(202) 296-5675
jband@policybandwidth.com

Counsel for Amicus Curiae

November 7, 2014

i

TABLE OF CONTENTS

TABLE OF CONTENTS – Continued

iii

TABLE OF AUTHORITIES

TABLE OF AUTHORITIES – Continued

Page

STATUTES

TABLE OF AUTHORITIES – Continued

TABLE OF AUTHORITIES – Continued

TABLE OF AUTHORITIES – Continued

TABLE OF AUTHORITIES – Continued

INTEREST OF *AMICUS CURIAE*[1]

The Computer & Communications Industry Association ("CCIA") represents over twenty companies of all sizes providing high technology products and services, including computer hardware and software, electronic commerce, telecommunications, and Internet products and services – companies that collectively generate more than $465 billion in annual revenues.[2] CCIA members have a large stake in the rules of software copyright being properly designed: effective intellectual property protection encourages developers to create new applications, but the improper extension of copyright law to functional elements will discourage innovation and inhibit competition in the industry.

For more than twenty-five years, CCIA has supported interpreting the intellectual property laws to permit the development of compatible products. For example, CCIA filed an *amicus* brief with this Court in *Lotus Dev. Corp. v. Borland Int'l*, 516 U.S. 233 (1996), arguing that the First Circuit's conclusion that copyright protection did not extend to the Lotus

---

[1] No counsel for any party authored this brief in whole or part, and no person other than *amicus curiae* or its counsel made a monetary contribution to the preparation or submission of this brief. All parties received timely notice of CCIA's intent to file, and consented to the filing of this brief.

[2] A list of CCIA members is available at https://www.ccianet. org/members. Google is a CCIA member, and Oracle and Sun were formerly members of CCIA, but none of these parties took any part in the preparation of this brief.

1-2-3 command structure followed this Court's decision in *Feist Publ'ns, Inc. v. Rural Tel. Serv. Co.*, 499 U.S. 340 (1991). CCIA also submitted an *amicus* brief in support of Google in the Federal Circuit.

In its Petition, Google asks this Court to review the Federal Circuit's disregard of longstanding principles concerning the scope of copyright protection for computer programs in favor of a discredited thirty-year-old Third Circuit decision, *Apple Computer v. Franklin Computer*, 714 F.2d 1240, 1253 (3d Cir. 1983) (stating that compatibility is "a commercial and competitive objective which does not enter into the somewhat metaphysical issue of whether particular ideas and expression have merged"). Adopting the position that copyright protects the elements of computer programs necessary to achieve compatibility poses serious anticompetitive consequences for CCIA members and the technology industry as a whole.

SUMMARY OF ARGUMENT

Before the Federal Circuit, Oracle repeatedly compared the Java Application Program Interface ("API") to a Harry Potter novel. Brief of Plaintiff-Appellant, at 1, 4, 13, 42-44, 52, 58, *Oracle America, Inc. v. Google Inc.*, Dkt. No. 43, Nos. 2013-1021, *et al.* (Fed. Cir. Feb. 11, 2013). In siding with this inapt comparison, the Federal Circuit departed from a point of broad, international consensus. Software applications, copyright has long acknowledged, are

different from artistic works. A novel stands by itself; a software application can function only in conjunction with hardware and other software, including an operating system. However, the word processor cannot run on every operating system. It can run only on an operating system with which it is "compatible," that is, one which conforms to the same set of technical rules.[3] These rules, known as "interface specifications," have no analog in the Harry Potter scenario.

The specific elements of the Java Application Program Interface ("API") at issue in this case are an example of interface specifications. In many cases such as this, "late-arriving hardware or software producers must, to compete, make their products compatible with the products sold by entrenched industry leaders, an effort that will characteristically require them to copy the industry leaders' interface specifications – the key that opens the lock to their operating systems." 2 GOLDSTEIN ON COPYRIGHT § 8.5.1 (2d ed. 2005). If a company could exercise proprietary control over the interface specifications implemented by its products, that company could determine which products made by other firms – if any – would be compatible with its software. And should that company have a dominant position in a particular market, it could use its control over compatibility to expand its dominant position into adjacent markets. Moreover,

---

[3] In this brief, the term "compatibility" is used interchangeably with "interoperability."

such authority would extend the rights under copyright beyond what is necessary to protect the original expressive elements that have traditionally been offered protection under American copyright law, and it would override limitations on copyright crafted to protect the public good.

Such a broad monopoly would have serious implications for consumer welfare. In the absence of competition during the effective lifespan of the product, the first developer would have little incentive to develop more innovative and less costly products. These negative consequences would be compounded by the fact that the personal computer revolution and the emergence of the Internet have produced an overwhelming need for interconnection between different elements of computer systems. Prohibiting competitors from accessing *de facto* standard interface specifications would lock users into a particular operating system or network software environment, and would inhibit the transfer of data between users with different computing environments. *See Lotus Dev. Corp. v. Borland Int'l*, 49 F.3d 807, 821 (1st Cir. 1995) (Boudin, J., concurring), *aff'd by an equally divided court*, 516 U.S. 233 (1996). The Petition shows a host of real-world problems and economic harms that would result if API copyright could foreclose compatibility, including the cost of rewriting interface code formerly understood to be unprotected, and lock-in costs resulting from consumers' inability to switch operating systems or cloud computing providers. Brief of Petitioner at 33-36. Lock-in would

deter competition, investment, and innovation in the burgeoning cloud computing industry, which is known to be sensitive to policy changes in copyright.[4]

In short, in the computer industry, overly broad intellectual property protection directly restricts competition and innovation. This was the status quo in the computing environment in the 1970s. Once a buyer purchased a computer system, the buyer was essentially locked-in to that system: the system was incompatible with products manufactured by other companies, and conversion costs were high. Although "locking in" was extremely profitable for dominant vendors such as IBM, competitors and users suffered from high prices, indifferent service, limited choice, and slow innovation. JONATHAN BAND & MASANOBU KATOH, INTERFACES ON TRIAL 2.0 at 1 (2011).

Google is correct that the courts of appeals are in disarray about the application of Section 102(b) to software. But until the Federal Circuit embraced what had been sidelined as a minority, discredited view, there was consensus outside of the Third Circuit on the narrower question of the protectability of program elements necessary for compatibility. As the district court explained in detail, courts outside the Third Circuit consistently rejected the *dicta* in *Franklin*

---

[4] Josh Lerner, *The Impact of Copyright Policy Changes on Venture Capital Investment in Cloud Computing Companies* (2011), *available at* http://www.analysisgroup.com/uploaded Files/Publishing/Articles/Lerner_Fall2011_Copyright_Policy_VC_ Investments.pdf.

and holding in *Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc.*, 797 F.2d 1222 (3d Cir. 1986), and found instead that interface specifications fall on the idea (or unprotected) side of copyright's idea/expression dichotomy. These more recent rulings have enabled the transition from the locked-in computer environments of the 1970s to today's interoperable Internet. The Federal Circuit, by adopting the Third Circuit's approach to compatibility, threatens this dynamic and innovative ecosystem.

This brief will not repeat Google's substantive arguments. Instead, given that "a page of history is worth a volume of logic," *eBay v. MercExchange*, 547 U.S. 388, 394 (2006) (Roberts, J., concurring) (citation omitted), this brief discusses how U.S. courts, Congress, and jurisdictions worldwide had arrived at a consensus interpretation of the copyright question of compatibility over the past 25 years, a consensus that the Federal Circuit's decision has disrupted.[5] This pro-compatibility interpretation of copyright law consists of two independent, but related principles. The first principle is that copyright protection does not extend to program elements necessary for compatibility,

---

[5] This history is discussed in detail in two books co-authored by counsel of record on this brief. JONATHAN BAND & MASANOBU KATOH, INTERFACES ON TRIAL: INTELLECTUAL PROPERTY AND INTEROPERABILITY IN THE GLOBAL SOFTWARE INDUSTRY (1995), *available at* http://tinyurl.com/InterfacesonTrial (hereinafter "INTERFACES 1.0"); and BAND & KATOH, INTERFACES ON TRIAL 2.0 (2011), *available at* http://mitpress.mit.edu/books/interfaces-trial-20 (hereinafter "INTERFACES 2.0").

such as interface specifications. The second principle, not directly contested in this litigation, is that the copying incidental to the reverse engineering necessary to identify these interface specifications does not infringe copyright.

Courts in four circuits (including the Federal Circuit) have concluded that reverse engineering is permissible when done to discover interface specifications necessary for compatibility. Yet Oracle now contends that interoperating with such specifications is itself infringement. If this is so, then these numerous appellate court decisions blessing reverse engineering when done in pursuit of compatibility become meaningless rulings. These cases were predicated upon the universally agreed-upon notion that copyright protection did not extend to interface specifications necessary for compatibility – a principle to which Oracle and Sun Microsystems, which created Java, both subscribed.

Following *Franklin* and *Whelan*, Sun and Oracle, along with a large swath of the computer industry, encouraged a pro-compatibility interpretation of copyright law.[6] This perspective was embodied in political advocacy and in *amicus* briefs in at least 16

---

[6] CCIA, the American Committee for Interoperable Systems ("ACIS"), and leading members of both organizations, ranging from Amdahl to Zenith, and including Oracle and Sun Microsystems, were active in promoting pro-compatibility rules. Sun's Deputy General Counsel Peter Choy chaired ACIS for much of its existence.

different cases. This coalition also advocated in favor of pro-compatibility principles in the Digital Millennium Copyright Act ("DMCA"), which are now mandated by eleven U.S. free trade agreements. The issue was so foundational to software development that industry advocacy for compatibility reached Europe, the Pacific Rim, and the Middle East. Thus, the Federal Circuit's decision not only rekindles a split between circuits by embracing a long discredited holding; it also puts a minority of U.S. courts at odds with an established international consensus.

Accordingly, the Petition should be granted to prevent the Federal Circuit from turning back the clock on a quarter century of established domestic and international software copyright jurisprudence.

ARGUMENT

I. THE FEDERAL CIRCUIT'S DECISION UPSETS AN INTERNATIONAL CONSENSUS THAT COPYRIGHT MUST NOT INTERFERE WITH COMPATIBILITY.

A. Modern Copyright Law Encourages Software Compatibility.

Over the past 25 years, U.S. courts, the Congress, and foreign jurisdictions have repeatedly applied copyright law in a manner that supports compatibility. Decision-makers around the world have adopted two related principles to this end. First, they have determined that copyright does not protect interface specifications and other program elements

necessary for compatibility. Second, they have refused to treat as copyright infringement any reproductions performed in the course of the reverse engineering necessary to discern these interface specifications.

The first principle – the non-protectability of interface specifications – was directly addressed by the district court and Google's Petition. The critical turning point in the development of this principle occurred in 1992 when the Second Circuit in *Computer Assocs. Int'l v. Altai, Inc.*, 982 F.2d 693 (2d Cir. 1992), followed this Court's 1991 decision in *Feist* to reject the Third Circuit's 1986 *Whelan* decision and 1983 *Franklin* decision.

Informed by this Court's holdings in *Feist* that the scope of copyright protection in utilitarian works is "thin," *Feist*, 499 U.S. at 350, and that substantial effort cannot confer copyright protection on unprotectable elements, *id*. at 349, the Second Circuit found that *"Feist* implicitly undercuts the *Whelan* [incentive based] rationale," *Altai*, 982 F.2d at 711, and ruled that under Section 102(b), copyright did not extend to program elements necessary for compatibility. By the end of the millennium, "most courts of appeal that ha[d] subsequently addressed the question of scope of copyright protection for computer programs ha[d], like *Computer Associates*, effectively rejected the *Whelan* approach." 1 GOLDSTEIN ON COPYRIGHT § 2.15.1 (2d ed. 2005) (citing *Sega Enters. v. Accolade, Inc*., 977 F.2d 1510 (9th Cir. 1992); *Mitel, Inc. v. Iqtel, Inc.*, 124 F.3d 1366 (10th Cir. 1997); and *Atari Games Corp. v. Nintendo of America,* 975 F.2d

832 (Fed. Cir. 1992)); *see also Lexmark Int'l v. Static Control Components, Inc.*, 387 F.3d 522, 534 (6th Cir. 2004).[7]

The second principle – the permissibility of reverse engineering – is not directly at issue in this case, but its treatment reflects on the importance that U.S. copyright law attributes to compatibility. Because a program's interface specifications usually are not readily apparent, developers seeking to interoperate often must research the interface specifications of the original program. This research, known as reverse engineering, is a basic tool of software product development. Without it, compatibility can be difficult or impossible to achieve.[8]

Copyright law, however, could impede software reverse engineering, because reverse engineering

---

[7] The Third Circuit, in contrast, continued the wayward trend of *Franklin* and *Whelan* in *Dun & Bradstreet Software Servs. v. Grace Consulting, Inc.*, 307 F.3d 197, 216 (3d Cir. 2002).

[8] This Court has long recognized that there is nothing inherently wrong with studying a competitor's product to understand how it works and to figure out how to make a better product. Thus, in *Kewanee Oil Co. v. Bicron Corp.*, 416 U.S. 470, 476 (1974), the Court stated that "trade secret law . . . does not offer protection against discovery by fair and honest means, such as . . . by so-called reverse engineering, that is by starting with a known product and working backward to divine the process which aided in its development or manufacture." The Court has also recognized the benefits of reverse engineering: "Reverse engineering . . . often leads to significant advances in technology." *Bonito Boats, Inc. v. Thunder Craft Boats, Inc.*, 489 U.S. 141, 160 (1989).

almost always requires making a reproduction or derivative work. At least six U.S. courts have held that reproduction that occurs while reverse engineering in order to learn interface specifications is permitted fair use. *See, e.g.*, *Atari*; *Sega* (citing 17 U.S.C. § 107).[9] Other courts have prevented enforcement under a copyright misuse theory. *See, e.g.*, *Alcatel U.S.A. v. DGI Techs.*, 166 F.3d 772 (5th Cir. 1999). Why would decades of copyright law permit reverse engineering if using the information learned by doing so – interface specifications – infringed copyright? The widely accepted principle that reverse engineering to learn essential interface specifications is not infringement only makes sense in a copyright system where utilizing those same interface specifications to interoperate is similarly non-infringing conduct.

In *Sega*, for example, the Ninth Circuit excused the copying incidental to Accolade's reverse engineering of Sega's products because the purpose of the reverse engineering was to uncover the unprotectable interfaces specifications necessary to achieve compatibility. Unfortunately, the Federal Circuit misunderstood *Sega*'s teaching. Because *Sega* was a fair use case, the Federal Circuit concluded that compatibility was

---

[9] *See also DSC Comms. Corp. v. DGI Techs.*, 898 F. Supp. 1183 (N.D. Tex. 1995), *aff'd*, 81 F.3d 597 (5th Cir. 1996); *Bateman v. Mnemonics, Inc.*, 79 F.3d 1532 (11th Cir. 1996); *DSC Comms. Corp. v. Pulse Comms. Inc.*, 976 F. Supp. 359 (E.D. Va. 1997), *aff'd in part, rev'd in part, and vacated in part*, 170 F.3d 1354 (Fed. Cir. 1999); *Sony Computer Entm't v. Connectix Corp.*, 203 F.3d 596 (9th Cir. 2000).

relevant only to the question of the applicability of the fair use defense.

To the contrary, the *Sega* court found that Accolade reverse engineered "Sega's software solely to discover the functional requirements for compatibility with the Genesis console – aspects of Sega's programs that are not protected by copyright. 17 U.S.C. § 102(b)." *Sega*, 977 F.2d at 1522. The Ninth Circuit explained that if reverse engineering were not permitted,

> the owner of the copyright gains a de facto monopoly over the functional aspects of his work – aspects that were expressly denied copyright protection by Congress. 17 U.S.C. §102(b). In order to enjoy a lawful monopoly over the idea or functional principle underlying a work, the creator must satisfy the more stringent standards imposed by the patent laws.

Id. at 1526. Indeed, *Sega*'s repeated citation to Section 102(b) leaves no doubt that the Ninth Circuit believed that elements necessary for compatibility were unprotectable under Section 102(b) – not that they were protectable but nonetheless could be copied as a matter of fair use.[10]

---

[10] The Ninth Circuit's subsequent decision in *Sony v. Connectix*, 203 F.3d at 603, that fair use permitted the reverse engineering by Connectix, was likewise predicated on the unprotectability of the interface information Connectix was seeking.

B. The Consensus About Software Compatibility Has Developed into an International Norm Over Time.

The current consensus that copyright should not inhibit compatibility did not arise in a vacuum. *Franklin* and *Whelan* aroused concerns in the computer industry that large incumbents might use copyright anticompetitively to restrict access to interface specifications and impede software compatibility. This led to an extensive debate, which continued until courts and policymakers coalesced around the principles reflected in *Lotus v. Borland*.

In December 1991, new entrants in the hardware and software sectors convened at Sun's headquarters to discuss these concerns. Chaired by Sun's Deputy General Counsel Peter Choy, this group – the American Committee for Interoperable Systems ("ACIS") – agreed upon a Statement of Principles, chiefly, that "[t]he rules or specifications according to which data must be organized in order to communicate with another program or computer, i.e., interfaces and access protocols, are not protectable expression under copyright law," and that copyright does not "restrict the ability of others to reproduce all or part of a lawfully obtained program as a step in the

development of competing products. . . ."[11] Similar efforts arose in other countries around the world.[12]

1. Pro-Compatibility Advocacy in the United States

Amicus CCIA, ACIS, and others in the industry advocated in support of copyright principles favorable to compatibility in numerous cases in the lower courts, arguing, as they had in *Lotus v. Borland*, that "extending copyright protection to elements necessary for interoperability would inhibit the ability of [industry] to develop innovative, competitive products."[13] In addition to cases involving the non-protectability of interface specifications, industry leaders also

---

[11] ACIS, Statement of Principles (1991), *available at* https://www.ccianet.org/interop.

[12] Joining CCIA and ACIS in this global effort were the European Committee for Interoperable Systems ("ECIS"), the Canadian Association for Interoperable Systems ("CAIS") and the Supporters of Interoperable Systems in Australia ("SISA"), all of whom subscribed to the position that copyright should not extend to interface specifications. *See* INTERFACES 1.0, *supra* n.5. Both Oracle and Sun were CCIA members at this time, as well as members of ACIS, ECIS, and SISA. *Id.* at 308. Sun joined CCIA in 1993 and remained a member until its 2010 acquisition by Oracle. Oracle was a member of CCIA from 1993 until 2011. Google joined CCIA in 2006, decades after CCIA's pro-compatibility advocacy began.

[13] Brief for ACIS & CCIA as *Amici Curiae* Supporting Respondent, *Lotus v. Borland*, 516 U.S. 233 (1995) (No. 94-2003) (submitted by Counsel of Record Peter M.C. Choy, Deputy Gen. Counsel, Sun; Chairman, ACIS).

championed the permissibility of software reverse engineering, and the interoperability exception of the DMCA.

The consensus of U.S. courts outside the Third Circuit precipitated a change in U.S. domestic and foreign policy. The U.S. Government eventually took the position that interface specifications should not receive copyright protection, including in its antitrust case against Microsoft. The Justice Department had objected to certain restrictions in licensing agreements, and, citing *Altai*, argued that copyright is not an unbounded property right, but rather a limited power designed to incentivize creation. The Government stated, "it is by now well established that the copyright in a computer program cannot extend to the functional aspects of that computer program; to design choices dictated by necessity, cost, convenience or consumer demand." Response of the United States to Microsoft's Motion for Summary Judgment, *U.S. v. Microsoft*, 1998 U.S. Dist. LEXIS 14231 (D.D.C. Sept. 14, 1998), at 77. To support this statement, it summarized *Mitel v. Iqtel*, 124 F.3d at 1366, as follows: "interface specifications of a communications protocol are freely copiable because they are functional rather than expressive." U.S. Response at 79.

Compatibility also found support elsewhere in the U.S. Government, as the Federal Trade Commission expressed concern in relation to Article 2B of the Uniform Commercial Code, insofar as it could limit the reverse engineering permitted under *Sega*, and

thereby dampen competition in the software industry. *See* INTERFACES 2.0, *supra* n.5, at 67-70.

Compatibility likewise found support in Congress, when it adopted the interoperability exception in the DMCA, 17 U.S.C. § 1201(f). This exception was included in response to lobbying by CCIA and ACIS.[14]

2. Free Trade Agreements Mandate Protections for Compatibility

Pro-compatibility principles also influenced the contours of U.S. trade agreements. Since 2002, U.S. free trade agreements ("FTAs") have included provisions modeled on the interoperability exception to Section 1201 of the DMCA. *See, e.g.*, U.S.-Korea Free Trade Agreement, art. 18.4.7(d)(i), June 30, 2007, 46 I.L.M. 642; (parties may permit "[n]oninfringing reverse engineering activities with regard to a lawfully obtained copy of a computer program . . . for the sole purpose of achieving interoperability of an independently created computer program with other programs."). Interoperability exceptions appear in

---

[14] In a 1998 press release, Michael Morris, then Vice President and General Counsel of Sun Microsystems, argued that the DMCA as introduced would "impose[ ] a new and unnecessary layer of restraint on lawful access to those unprotected elements of computer programs that are necessary to achieve interoperability, thus placing developers of interoperable products at the mercy of proprietary vendors." Press Release, Sun Microsystems, House IP Subcommittee Action Threatens Internet Competition (Mar. 1, 1998).

FTAs with Australia, Bahrain, Chile, Colombia, Costa Rica, Dominican Republic, El Salvador, Guatemala, Honduras, Morocco, Nicaragua, Oman, Panama, Peru, and Singapore.[15] As in the United States, many of these countries have adopted reverse engineering exceptions in their domestic law.

C. Copyright Laws Around the World Protect Compatibility.

In addition to the reverse engineering exceptions adopted pursuant to the FTAs, legislation favoring compatibility has been adopted in over 40 countries, including many major U.S. trading partners.

---

[15] *See* U.S.-Chile Free Trade Agreement art. 17.7.5(d)(ii), June 6, 2003, 42 I.L.M. 1026; U.S.-Dominican Republic-Central America Free Trade Agreement art. 15.57(d)(ii), May 28, 2004, 43 I.L.M. 514; U.S.-Australia Free Trade Agreement art. 17.4.7(e)(i), May 18, 2004, 43 I.L.M. 1248; U.S.-Morocco Free Trade Agreement, art. 15.5.8(d)(i), June 15, 2004, 44 I.L.M. 544; U.S.-Bahrain Free Trade Agreement art. 14.4.7(e)(i), Sept. 14, 2004, 44 I.L.M. 544; U.S.-Peru Trade Promotion Agreement art. 16.7.4(e)(i), Apr. 12, 2006; U.S.-Colombia Trade Promotion Agreement art. 16.7.4(e)(i), Nov. 22, 2006; U.S.-Singapore Free Trade Agreement art. 16.4.7(e)(i), May 6, 2003, 42 I.L.M. 1026; U.S.-Panama Trade Promotion Agreement, art. 15.5.7(d)(i), June 28, 2007; U.S.-Oman Free Trade Agreement, art. 15.4.7(d)(i), Jan. 1, 2009.

1. European Union Law Mirrors the U.S. Pro-Compatibility Approach

In 1991, the European Union adopted a Software Directive, which reflects a policy judgment that copyright should not interfere with interoperability.[16] Council of Ministers Directive 91/250/EEC of 14 May 1991 on the Legal Protection of Computer Programs, 1991 O.J. (L 122). In particular, Article 6 of the Software Directive permits reverse engineering "indispensible to obtain the information necessary to achieve . . . interoperability." The Software Directive has been implemented by all member states of the EU, as well as Norway, Russia, Serbia, Switzerland, and Turkey. INTERFACES 2.0, *supra* n.5, at 6. While the Software Directive does not directly address the protectability of interface specifications, commentators generally perceived that "the law on software copyright interoperability issues seem[ed] quite settled on both sides of the Atlantic,"[17] and that interface information necessary to achieve interoperability must fall on the "idea" side of the idea/expression

---

[16] The legislative process leading to the adoption of the Directive, including the participation of ECIS (whose members included Oracle and Sun), is discussed in detail in INTERFACES 1.0, *supra* n.5, at 227-41.

[17] Pamela Samuelson, *The Past, Present, and Future of Software Copyright Interoperability Rules in the European Union and United States*, 34(3) EUR. INTELL. PROP. REV. 229 (2010).

dichotomy.[18] The specific issue received scant attention until May 2012, when the EU's highest court ruled in *SAS Institute v. World Programming*[19] that program functionality, programming languages, and data formats were not protectable under the Software Directive. The case concerned SAS Institute's allegations that World Programming's effort to create "middleware" which interoperated with SAS software. SAS claimed this infringed its copyrights, *inter alia*, by replicating (i) the SAS programming language, (ii) the data and programming interfaces used in the SAS system, and (iii) the functionality offered by the SAS System.

The CJEU held that the Software Directive "must be interpreted as meaning that neither the functionality of a computer program nor the programming language and the format of data files used in a computer program in order to exploit its functions constitute a form of expression of that program and, as such, are not protected by copyright. . . ." *Id*. ¶ 40. The CJEU observed that "the main advantage of protecting computer programs by copyright" as opposed, presumably, to patents, "is that such protection covers only the individual expression of the work and thus leaves other authors the desired latitude to

---

[18] Article 1(2) provides that "[i]deas and principles which underlie any element of a computer program, including those which underlie its interfaces, are not protected by copyright. . . ."

[19] *SAS Institute, Inc. v. World Programming Ltd.*, Case C-406/10, [2012] 3 CMLR 4.

create similar or even identical programs," *id*. ¶ 41, provided that they refrain from copying protected expression. In other words, the CJEU reached precisely the same conclusion as the district court below, and the opposite of the Federal Circuit.

2. Copyright Policies Around the Pacific and Across the World Align with U.S. and European Pro-Compatibility Law

Policymakers in the Pacific Rim have also arrived at a view consistent with that of the U.S. and Europe. During a decade-long copyright law review in Australia, SISA filed numerous submissions on behalf of its members, including Sun and Oracle, in support of an exception for reverse engineering for purposes of compatibility, *see* INTERFACES 2.0, *supra* n.5, at 136-58, which Australia ultimately adopted. Australian officials explained that the government's rationale for introducing these exceptions was to align itself with leading nations, including the U.S. and Europe. With the advent of the Internet, "there is an obvious need for computers and the programs which drive them to communicate, connect, or 'interoperate' with each other." *Id.* at 152.[20]

---

[20] Quoting from Speech of Hon. Daryl Williams QC, on Copyright Amendment (Computer Program) Bill 1999, Second Reading (Aug. 11, 1999).

A similar discussion occurred in Hong Kong during a copyright revision in the months before the 1997 turnover to China. *See* INTERFACES 2.0, *supra* n.5, at 168-75. After hearing testimony in favor of reverse engineering (including from Sun), the Legislative Council broadened Hong Kong's fair dealing provision to more closely resemble the fair use provision of the U.S. Copyright Act, in order "to encourage competition in the information technology industry by facilitating timely access to information and ideas underlying computer programs." *Id.* at 175.[21]

Over the following years, other Pacific Rim countries, including Malaysia,[22] New Zealand,[23] the Philippines,[24]

---

[21] Quoting from Speech by Secretary of Trade and Industry on Resumption of Second Reading, Debate at 10 (June 24, 1997).

[22] Copyright (Amendment) Act 2012, Laws of Malaysia, Act A1420, § 36A(2)(a) (exception applying where "the sole purpose of achieving interoperability of an independently created computer program with the original program or any other programs.").

[23] New Zealand Copyright (New Technologies) Amendment Act 2008, § 43 (amending Copyright Act 1994, § 80A(2)) (2008) (permitting decompilation "necessary to obtain information necessary for the objective of creating an independent program that can be operated with the program decompiled or with another program. . . .").

[24] "Decompilation, which is the reproduction of code and translation of the form of the computer program indispensable to obtain the information necessary to achieve the interoperability of an independently created computer program with other programs may also constitute fair use." INTERFACES ON TRIAL 2.0, *supra* n.5, at 178 (quoting Intellectual Property Code of the Philippines, § 185.1).

Singapore,[25] and Taiwan[26] all amended their copyright laws to encourage software compatibility, often citing the U.S. approach.[27]

Various other nations worldwide have also explicitly embraced interoperability in their copyright statutes, including India,[28] Canada,[29] Kenya,[30] and

---

[25] Singapore's amendment aimed to "bring [it] in line with the United States, the United Kingdom, other European Union countries, Hong Kong, and Australia, which do not bar the use of copyright materials for commercial research." INTERFACES 2.0, at 158-67 (quoting Attorney-General of Law, Second Reading of Copyright (Amendment) Bill of 1998 (Sing.) (February 19, 1998)).

[26] Copyright Law of the Republic of China, art. 65, Intellectual Property Office (2007) (Taiwan).

[27] ACIS also argued in favor of compatibility exceptions in Japan and Korea. *See* INTERFACES 1.0, *supra* n.5, at 297-316; INTERFACES 2.0, *supra* n.5, at 178-80.

[28] Copyright Act, No. 14 of 1957, India Code (1999), § 52(1)(ab) (permitting "the doing of any act necessary to obtain information essential for operating interoperability of an independently created computer programme with other programmes. . . .").

[29] Canada Copyright Modernization Act (Bill C-11), S.C. 2012, c. 20, s. 30.61 (permitting the owner or licensee of a copy of a computer program "to reproduce the copy for the sole purpose of obtaining information that would allow the person to make the program and any other computer program interoperable.") Further, the amendment permits the circumvention of a TPM "for the sole purpose of obtaining information that would allow the person to make the program and any other computer program interoperable." *See id.*, s. 41.12(1).

[30] Kenya Copyright Act, (2009) Cap. 130 § 26(5) (authorization "shall not be required to decompile [a] program, convert

Israel,[31] so as to align themselves with the "hard-fought peace and harmony on interoperability issues" that had until recently prevailed in the United States and Europe. *See* Samuelson, *supra* n.17, at 234.

In sum, U.S. courts, the Executive Branch, Congress, and over 40 countries have recognized that permitting copyright law to impede compatibility would harm legitimate competition in the computer industry and impede the growth of the Internet economy. By adopting the long discredited dicta of the Third Circuit in *Franklin*, the Federal Circuit's decision undermines a basic premise of the U.S. software industry: the elements necessary for compatibility do not receive copyright protection. This would stifle innovation in the United States, and place U.S. companies at a disadvantage to foreign competitors.

---

the program into a version expressed in different programming language, code, notation for the purpose of obtaining information needed to enable the program to operate with other programs.").

[31] Israel Copyright Act, 5767-2007, 2007 LSI 2199, § 24(c)(3) (permitting the copying of a computer program to "obtain[ ] information which is needed to adapt a different and independently developed computer system or program, in such a way that it will be interoperable with the computer program.").

II. THE FEDERAL CIRCUIT OVERPRO-TECTS COMPUTER PROGRAMS, TO THE DETRIMENT OF CONSUMERS, PRO-GRAMMERS, AND THE COMPUTER IN-DUSTRY.

The Federal Circuit claimed that Google's arguments "appear premised on the belief that copyright is not the correct legal grounds upon which to protect the intellectual property rights of software programs." Yet the Federal Circuit itself contradicted this assertion when it observed that

> Google agrees with the district court that the implementing code is the expression entitled to protection. . . . Indeed, at oral argument counsel for Google explained that, "it is not our position that none of Java is copyrightable. Obviously, Google spent two and a half years . . . to write from scratch all the implementing code."

Oracle America, Inc. v. Google Inc., 750 F.3d 1339, 1360 (Fed. Cir. 2014) ("*Oracle America II*"). As the Federal Circuit noted, "of the 37 Java API packages at issue, '97 percent of the Android lines were new from Google.'" *Id*. at 1352 (citation omitted).

The argument made by Google before the Federal Circuit, and now before this Court, is not that copyright does not protect computer programs. Rather, it is that copyright does not protect certain elements of programs, such as command structures and elements necessary for interoperability. The conclusion dictated by *Feist*, that copyright protection in computer

programs is "thin," is neither a value judgment nor a suggestion that engineers are less creative than novelists. Rather, it recognizes the basic fact that an applied work of engineering functions differently from a work of art or entertainment, and as a consequence, works of engineering receive a different scope of copyright protection.

Moreover, to say that interface specifications necessary for compatibility are not protected, and that software copyright is thin, is not to withhold copyright protection from software platforms. Title 17 still provides ample protection to many program elements without going through the effort of a patent prosecution, and offers robust remedies for copyright infringement in appropriate cases.[32]

The history of the computer industry is replete with examples of an established vendor using copyright law to lock-in consumers by preventing the development of innovative compatible products. The Federal Circuit attempts to distinguish this case in essence by asserting that the result of its ruling would not be to lock *consumers* into the Java environment, so much as to lock the community of Java *programmers* into the Java environment. It criticizes

---

[32] *See, e.g.*, Karen Gullo & Cornelius Rahn, *SAP to Pay Oracle $306 Million for Copyright Breach*, Bloomberg.com, Aug. 3, 2012, *available at* http://www.bloomberg.com/news/2012-08-02/oracle-says-sap-to-pay-306-million-in-copyright-deal.html. Copyright protection is available for lines of code, detailed structure, and graphical user interfaces.

Google for copying "Oracle's declaring code and SSO to capitalize on the preexisting community of programmers who were accustomed to using the Java API packages." *Oracle America II* at 1371. But locking programmers into a particular computing environment will have the same ultimate result: less consumer choice.

Further, preventing programmers from using their training and experience with Java on a different platform is antithetical to the goal of the copyright system to stimulate "creativity for the general public good." *Twentieth Century Music Corp. v. Aiken*, 422 U.S. 151, 156 (1975). The Federal Circuit opined that the desire to capitalize on the preexisting community of Java programmers "has nothing to do with copyrightability." *Oracle America II* at 1372. But this is wrong. What could be better proof that something is a procedure, system, or method of operation than if a person can become "trained," "experienced," or "accustomed" to using it in the course of developing new works? Earlier in its opinion, the Federal Circuit described the API packages as "shortcuts" programmers can use when writing their own programs. While the detailed steps of each shortcut may be copyrightable, the structure of the entire set of shortcuts surely isn't. If a framework of shortcuts used by programmers in their development process isn't a procedure, system, or method of operation, what is?

Since the early 1990s, computer programmers in the United States have understood that copyright

does not protect command structures and program elements necessary for interoperability. Based on this understanding, programmers have freely copied these elements, which has encouraged enormous creativity, innovation, and competition in the digital environment.

> Providing protection for such amorphous concepts as the overall structure of a program, without considering whether such a structure is protectable under traditional copyright theories, increases the risk of granting copyright holders protection on a par with that provided to patent holders. . . . Creative development in the software industry may well be stifled by overly broad copyright protection afforded to programs that represent the basic building blocks of a particular field.

4 DAVID NIMMER & MELVILLE B. NIMMER, NIMMER ON COPYRIGHT § 13.03[A][1][d] (2005).

The Federal Circuit's decision casts doubt on this understanding. By ruling that compatibility is relevant only to fair use, and not to protectability, the Federal Circuit would require a developer to perform a fair use analysis before developing a compatible product. Judge Boudin in the *Borland* case recognized that widespread application of the fair use doctrine for purposes of achieving compatibility "would entail a host of administrative problems that would cause cost and delay, and would also reduce the ability of the industry to predict outcomes." *Borland*, 49 F.3d at

821-22. This would place U.S. programmers at a competitive disadvantage to developers in other jurisdictions, such as the EU, that recognize that copyright does not protect program elements necessary for compatibility.

CONCLUSION

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

Respectfully submitted,

MATT SCHRUERS   JONATHAN BAND
ALI STERNBURG   *Counsel of Record*
COMPUTER & COMMUNICATIONS   JONATHAN BAND PLLC
    INDUSTRY ASSOCIATION   21 Dupont Circle NW,
900 17th Street NW, Suite 1100   Suite 800
Washington, D.C. 20006   Washington, D.C. 20036
(202) 783-0070   (202) 296-5675
mschruers@ccianet.org   jband@policybandwidth.com

Counsel for Amicus Curiae

November 7, 2014