MORRISON & FOERSTER LLP
MICHAEL A. JACOBS (Bar No. 111664)
mjacobs@mofo.com
KENNETH A. KUWAYTI (Bar No. 145384)
kkuwayti@mofo.com
MARC DAVID PETERS (Bar No. 211725)
mdpeters@mofo.com
DANIEL P. MUINO (Bar No. 209624)
dmuino@mofo.com
755 Page Mill Road, Palo Alto, CA 94304-1018
Telephone: (650) 813-5600 / Facsimile: (650) 494-0792

BOIES, SCHILLER & FLEXNER LLP
DAVID BOIES (Admitted *Pro Hac Vice*)
dboies@bsfllp.com
333 Main Street, Armonk, NY 10504
Telephone: (914) 749-8200 / Facsimile: (914) 749-8300
STEVEN C. HOLTZMAN (Bar No. 144177)
sholtzman@bsfllp.com
1999 Harrison St., Suite 900, Oakland, CA 94612
Telephone: (510) 874-1000 / Facsimile: (510) 874-1460

ORACLE CORPORATION
DORIAN DALEY (Bar No. 129049)
dorian.daley@oracle.com
DEBORAH K. MILLER (Bar No. 95527)
deborah.miller@oracle.com
MATTHEW M. SARBORARIA (Bar No. 211600)
matthew.sarboraria@oracle.com
500 Oracle Parkway, Redwood City, CA 94065
Telephone: (650) 506-5200 / Facsimile: (650) 506-7114

Attorneys for Plaintiff
ORACLE AMERICA, INC.

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

SAN FRANCISCO DIVISION

ORACLE AMERICA, INC. Plaintiff, v. GOOGLE INC. Defendant.	Case No. CV 10-03561 WHA **ORACLE'S BRIEF ADDRESSING COURT'S COPYRIGHT QUESTIONS** Dept.: Courtroom 8, 19th Floor Judge: Honorable William H. Alsup

TABLE OF CONTENTS

Page

I. COURT'S QUESTIONS FROM APRIL 19, 2012 (RT AT 882-892) 1

A. Could you get a patent on the structure, sequence, and organization of the APIs? Is that the proper subject matter of a patent? 1

B. When software is registered with the Copyright Office, is the structure, sequence, and organization investigated by the Copyright Office? 1

C. For a derivative work, does the plaintiff have to prove that the defendant actually possessed and derived their own work from the copyrighted work? 2

D. Will the evidence show that the Google engineers who worked on Android source code possessed the Oracle Java API documentation? 2

E. Are there subsidiary questions of fact that the jury should decide that would tie into the judge's determination on copyrightability? 3

II. COURT'S QUESTIONS FROM APRIL 20, 2012 (RT AT 915-22) 3

A. Does the term "class libraries" refer only to the compiled object code or to the object code and the source code? 3

B. What are the copyrighted works? Did Oracle identify the copyrighted works in discovery responses? 4

1. Oracle Identified The Copyrighted Works In The Amended Complaint And In Discovery Responses 4

2. Registration Of The Different Versions Of The Java Platform Also Registered The Individual Works Within It 5

C. What will we tell the jury they should be comparing for copyright infringement purposes? 7

III. COURT'S QUESTIONS FROM ECF 948 8

A. What case law or other authority is there that states the judge must identify the "work as a whole" (for similarity, fair use, and de minimis) for the jury? Which party has the burden to identify the "entire work"? 8

B. With respect to what segment of the "work" can stand alone within the meaning of Hustler v. Moral, 796 F.2d 1148, 1155 (9th Cir. 1986), the Court wishes to know whether the implementation of the 37 API packages inherit, call upon, invoke, or incorporate any method, field, or class outside the 37 10

C. Why shouldn't we let the jury decide what the "work as a whole" is? 10

D. Is the "work as a whole" the same for purposes of "substantial similarity (or virtually identical)," "fair use," and "de minimis" copying? If not, how are the "works as a whole" to be found for these purposes? 10

E. For purposes of identifying the "work as a whole," should Oracle be held to the copyrighted work identified in the operative complaint? 11

F. If the SSO and declarations are held to be protected elements, then why are there still issues of access and similarity for purposes of infringement (excluding de minimis and fair use)? Put another way, isn't substantial similarity only an issue if there isn't an admission of factual copying of protectable elements? 13

TABLE OF CONTENTS
(continued)

Page

IV. COURT'S QUESTIONS FROM ECF 951 13

A. Do any of the Sun compiled lines in the 37 APIs call upon part or all of another API as a step? 13

B. If so, do the accused APIs likewise call upon the same other API? That is, even though the implementations are different from the Sun implementation, do the accused APIs borrow from other APIs in the same pattern? 14

C. If the answer to question A is yes, do any of the compiled lines in the 37 Sun APIs call upon part or all of another API outside the 37? Put differently, is all of the cross referencing done within the 37? 14

V. COURT'S QUESTIONS FROM ECF 953 14

A. What efficiencies, if any, are obtained by grouping methods or fields together under a single class? Put differently, what would be lost if a method that returned the cosine of an angle was grouped under a class other than Math? This discussion should get the pros and cons of the particular interrelationships (e.g., inheritance) within the 37 API packages 14

TABLE OF AUTHORITIES

Page(s)

CASES

AFL Telecomms. LLC v. SurplusEQ.com, Inc.,
2012 U.S. Dist. LEXIS 49239 (D. Ariz. Apr. 9, 2012) 12

Am. Geophysical Union v. Texaco, Inc. 5, 6

Apple Computer, Inc. v. Microsoft Corp.,
35 F.3d 1435 (9th Cir. 1994) 9

Atari Games Corp. v. Oman 1

Baxter v. MCA, Inc.,
812 F.2d 421 (9th Cir. 1987) 8

Bean v. Littell,
669 F. Supp. 2d 1031 (D. Ariz. 2008) 6

Bilski v. Kappos,
130 S. Ct. 3218 (2010) 1

Dr. Seuss Enters., L.P. v. Penguin Book USA, Inc.,
924 F. Supp. 1559 11

Fred Fisher, Inc. v. Dillingham,
298 F. 145 (S.D.N.Y.1924) (L. Hand, J.) 8

Home & Nature, Inc. v. Sherman Specialty Co.,
322 F. Supp. 2d 260 (E.D.N.Y. 2004) 11

Hustler Magazine, Inc. v. Moral Majority, Inc.,
796 F.2d 1148 (9th Cir. 1986) 6, 8, 10

L.A. Printex Indus., Inc. v. Aeorpostale, Inc.,
No. 10-56187, 2012 U.S. App. LEXIS 7079 (9th Cir. Apr. 9, 2012) 8

L.A. Times v. Free Republic,
1999 WL 33644483 (C.D. Cal. Nov. 8, 1999) 6, 7

Merch. Transaction Sys., Inc. v. Nelcela, Inc.,
2009 U.S. Dist. LEXIS 25663 (D. Ariz. Mar. 17, 2009) 9

Perfect 10, Inc. v. Amazon.com, Inc.,
508 F.3d 1146 (9th Cir. 2007) 9, 11

Perfect 10 v. Cybernet Ventures 11, 12

Range Road Music, Inc. v. East Coast Food, Inc.,
2012 U.S. App. LEXIS 3173 (9th Cir. Feb. 16, 2012) 9, 13

Reader's Digest Ass'n v. Conservative Digest, Inc. 1

Religious Tech. Ctr. v. Lerma,
1996 U.S. Dist. LEXIS 15454 (E.D.Va. Oct. 4, 1996) 6

Salestraq Am., LLC v. Zyskowski,
635 F. Supp. 2d 1178 (D. Nev. 2009) 12

Shaw v. Lindheim,
919 F.2d 1353 (9th Cir 1990) 7

Sheldon v. MGM,
81 F.2d 49 (2d Cir. 1936) 7

Smith v. Jackson,
84 F.3d 1213 (9th Cir. 1996) 2

Streetwise Maps v. VanDam, Inc.,
159 F.3d 739 (2d Cir. 1998) 12

Super Future Equities, Inc. v. Wells Fargo Bank Minn. 8, 9

STATUTES

35 U.S.C. § 101 1

OTHER AUTHORITIES

37 C.F.R. 202.3(b)(4)(i)(A) 5

37 C.F.R. § 202.3(b)(4)(i)(A) 5

Nimmer on Copyright § 13.03[F][5] 11

Oracle responds as follows to the Court's questions at trial (RT at 882-92, 915-22) and in recent orders (ECF Nos. 948, 951, 953) regarding copyright issues.

I. COURT'S QUESTIONS FROM APRIL 19, 2012 (RT at 882-892)

A. Could you get a patent on the structure, sequence, and organization of the APIs? Is that the proper subject matter of a patent?

No. The Patent Act specifies four categories of patent-eligible subject matter: processes, machines, manufactures, and matter compositions. *See* 35 U.S.C. § 101, *Bilski v. Kappos*, 130 S. Ct. 3218, 3225 (2010). Structure, sequence, and organization are none of these. Oracle's 4/3/11 Brief notes that patent-eligible subject matter may embody API elements. (ECF No. 853 at 6-7.)

B. When software is registered with the Copyright Office, is the structure, sequence, and organization investigated by the Copyright Office?

Whether the Copyright Office conducts such an investigation in any particular case is uncertain, and Oracle makes no claim here that such an investigation was conducted.

Decisional law relating to registration, however, indicates that the Copyright Office sometimes conducts such an examination and also provides useful guideposts in this case. In *Atari Games Corp. v. Oman*, the D.C. Circuit reversed the Copyright Office when it applied the wrong standard for copyrightability to the computer game BREAKOUT. 888 F.2d 878 (D.C. Cir. 1989). The defendant in *Atari* was the Register of Copyrights. The Register had determined that the simple shapes that made up the game's different elements did not themselves qualify for copyright protection and denied registration. *Id.* at 879–80. The D.C. Circuit reversed. The Register had improperly "subjected BREAKOUT to a component-by-component analysis," when its focus "ultimately should be" on the "total sequence of images displayed as the game is played." *Id*. at 883. Because "simple shapes, when selected or combined in a distinctive manner indicating some ingenuity, have been accorded copyright protection both by the Register and in court," the Register was bound to consider the arrangement of video game elements as a whole when determining whether or not to issue a certificate of copyright. *Id.* at 882–883.

Similarly, in *Reader's Digest Ass'n v. Conservative Digest, Inc.*, the D.C. Circuit held that the "distinctive arrangement and layout of" the ordinary lines, typefaces, and colors of the magazine meant that it should be "entitled to protection as a graphic work," because "Reader's

Digest has combined and arranged common forms to create a unique graphic design and layout." 821 F.2d 800, 806 (D.C. Cir. 1987). The court in *Atari* admonished the Register to "take careful account" of *Reader's Digest* in delivering its reconsidered opinion. *Atari*, 888 F.2d at 883.

These decisions warn against analyzing the API elements here on too granular a basis. As to names, for example, even if an individual name might not be protectable, the selection, arrangement, and structure of the names as embodied in API declarations, viewed as a whole, is copyrightable subject matter.

C. For a derivative work, does the plaintiff have to prove that the defendant actually possessed and derived their own work from the copyrighted work?

No. Oracle is required to prove that Google copied in order to prevail on a claim concerning a derivative work, but Oracle is aware of no case that affirmatively requires the plaintiff to show that an infringer actually had in its possession the work from which the derivative work is based. Even access (as opposed to possession) is not a necessary condition of an infringement case; proof of access lowers Oracle's burden on similarity. *See Smith v. Jackson*, 84 F.3d 1213, 1220 (9th Cir. 1996) ("access eliminates the need for a plaintiff to establish a 'striking similarity' between plaintiff's and defendant's work").

Here, however, Google has admitted copying. *See* section I.D, *infra*. If Oracle did not have this direct evidence of copying, it also would have been permitted to create an inference of copying by proving substantial similarity and access. Google, of course, has admitted both of these facts as well. (*See* ECF No 946 ("For the 37 accused API packages, Android and Java 2 SE Version 5.0 have substantially the same selection, arrangement and structure of API elements."); Google's Responses to Oracles RFAs ("Google admits that Android developers employed by Google had access to publicly available . . . documentation for Java 2 Standard Edition Version 5.0 before November 5, 2007.").)

D. Will the evidence show that the Google engineers who worked on Android source code possessed the Oracle Java API documentation?

Yes. Google has admitted such possession. Bob Lee, formerly the core libraries lead for Android, testified on Friday, April 20, that he consulted the Java API specifications while working on the core libraries:

Q. You consulted the Java API specifications to make sure that the Android code for the corresponding core libraries would be consistent with those specifications, correct?

A. Yes.

Q. The Java API specifications that you consulted were available on Sun's website, correct?

A. Yes.

Q. And you consulted those Java API specifications while you were doing work for Google on Android, correct?

A. Yes.

Q. You saw that there were copyright notices on the Java API specifications when you consulted them, correct?

A. Yes.

(RT at 982:25–983:12 (Lee).)

Counsel for Google also has admitted in open court that the Android engineers not only had access in the theoretical sense; they affirmatively looked at the Java API specifications while they were writing code for Android. (*See* RT at 910:23 –912:10 ("what Google used were the API specifications"); RT at 886:22–887:6 ("There was . . . in discovery, Your Honor, that at least some of the engineers who worked on the libraries did have access to, did look at, some of the English language prose descriptions of the APIs.").)

E. Are there subsidiary questions of fact that the jury should decide that would tie into the judge's determination on copyrightability?

No. There are no subsidiary questions of fact that the jury needs to decide relating to the judge's determination on copyrightability.

II. COURT'S QUESTIONS FROM APRIL 20, 2012 (RT at 915-22)[1]

A. Does the term "class libraries" refer only to the compiled object code or to the object code and the source code?

Used properly, "class library" refers only to the compiled object code. Mark Reinhold, Oracle's Chief Architect of the Java Platform Group, testified that a "library is the compiled form of the code that can be used directly." (RT at 592:17-593:3 (Reinhold).) One should use the

[1] To the extent the Court's questions of April 20, 2012, were subsumed within the questions posed in ECF No. 948, they have been omitted in this section.

phrases "class library source code" or "source code for the class libraries" to refer to the source code. Oracle's counsel has sometimes not observed the distinction between source code and object code. (*See, e.g.*, ECF No. 859 at 3 ("The specifications are documents, the class libraries are source code programs."); *but see* ECF No. 645 at 2 ("Class libraries are collections of already-compiled classes.").)

B. What are the copyrighted works? Did Oracle identify the copyrighted works in discovery responses?

1. Oracle Identified The Copyrighted Works In The Amended Complaint And In Discovery Responses

The copyrighted works at issue are the APIs for the 37 packages and their associated class libraries (and their associated source code) and the 11 individual computer program code files. These were encompassed within the copyright registration for J2SE 5.0 and J2SE 1.4, which were registered as collective works, and also in various earlier registrations identified in the registrations for J2SE 5.0 and J2SE 1.4.

Oracle identified these copyrighted works in the Amended Complaint and in discovery. The Amended Complaint states:

> Android includes infringing class libraries and documentation. Approximately one third of Android's Application Programmer Interface (API) packages (available at http://developer.android.com/reference/packages.html) are derivative of Oracle America's copyrighted Java API packages (available at http://download-llnw.oracle.com/javase/1.5.0/docs/api/ and http://download-llnw.oracle.com/javase/1.4.2/docs/api/) and corresponding documents. The infringed elements of Oracle America's copyrighted work include Java method and class names, definitions, organization, and parameters; the structure, organization and content of Java class libraries; and the content and organization of Java's documentation. Examples of this copying are illustrated in Exhibit I to this complaint.

(ECF No. 36 ¶ 40.) The referenced Exhibit I includes the API specification documentation with which the Court is now familiar for both Java and Android for the Security class, located within the package java.security, one of the 37 API packages accused in this case. (ECF No. 36-9.)

This paragraph continues: "In at least several instances, Android computer code also was directly copied from copyrighted Oracle America code." (ECF No. 36 ¶ 40.) It specifically identifies "PolicyNodeImpl.java", one of the 11 copied files, and attaches it as Exhibit J. (ECF No. 36-10.) At that time, Oracle had not identified all of the literally copied files.

Oracle then identified the infringed packages in interrogatory responses served in January 2011. In its response to Interrogatory No. 2, Oracle identified 51 packages, attaching further side-by-side comparisons of the Java and Android APIs for several packages. (Ex. A at 7-9 (Oracle Resp. to Interrogatory No. 2).) That same response identified the 12 accused Android files. (*See id.*) The number of infringed packages was later reduced to 37 in supplemental interrogatory responses served in July 2011.

2. Registration Of The Different Versions Of The Java Platform Also Registered The Individual Works Within It

Google misstated the law regarding copyright registration and the Court's prior ruling on this issue. The Court correctly recalled that it resolved this issue against Google. (*See* RT 912:25-913:7.) Oracle registered the versions of the Java platform as collective works under a single copyright registration. This is permitted under the plain language of 37 C.F.R. § 202.3(b)(4)(i)(A).[2] The Court held that Google's proposed interpretation of the regulation was incorrect:

> The plain meaning of this provision is that when a single published unit contains multiple elements "that are otherwise recognizable as self-contained works," the unit is considered a single work *for the limited purpose of registration* while its elements may be recognized as separate works for other purposes.

(ECF No. 433 at 6 (emphasis in original).)

This principle is well supported by case law. In *Am. Geophysical Union v. Texaco, Inc.*, the court held that each article within a journal was protected by copyright even though the publisher chose to register only each journal with the Copyright Office. 802 F. Supp 1, 17 (S.D.N.Y. 1992). The court rejected the defendant's argument that the work as a whole should be the journal that was registered:

> This argument constitutes imaginative lawyering, but it does not prevail. Each article, note or letter published in *Catalysis* is a separately authored work, protected by a copyright, which the authors have assigned to Academic Press. Because it would

[2] That provision states: "For the purpose of registration on a single application and upon payment of a single registration fee, the following shall be considered a single work: (A) In the case of published works: all copyrightable elements that are otherwise recognizable as self-contained works, that are included in a single unit of publication, and in which the copyright claimant is the same." 37 C.F.R. 202.3(b)(4)(i)(A).

> involve gigantic expense and inconvenience to register separately each of the 20 odd items that appear in an individual issue, Academic Press registers each issue with the Copyright Office. It does not follow from the manner of registration with the Copyright Office that the "copyrighted work" for the purposes of fair use analysis consists of the entire issue rather than the separate creations of the separate authors.

Id. at *17.

The Ninth Circuit reached a similar conclusion in *Hustler*, finding that the "entire work" there consisted of a one-page advertisement parody in a 154-page magazine. *Hustler Magazine, Inc. v. Moral Majority, Inc.*, 796 F.2d 1148, 1154 (9th Cir. 1986) ("the parody is not an interwoven component of the magazine, but can stand totally alone. A creative work does not deserve less copyright protection just because it is part of a composite work."); *see also Religious Tech. Ctr. v. Lerma*, 1996 U.S. Dist. LEXIS 15454, at *27 (E.D.Va. Oct. 4, 1996) ("Although Lerma did not post the entirety of [the materials registered with the Copyright Office], he did post the entirety of certain discrete subparts of these series. Under the Code of Federal Regulations and under case law, these subparts constitute single works and are the benchmark against which to compare Lerma's actions."); *Bean v. Littell*, 669 F. Supp. 2d 1031, 1034 (D. Ariz. 2008) ("When a claimant registers a collective work, the copyright protection can also extend to each constituent part of that work."); *L.A. Times v. Free Republic*, 1999 WL 33644483, at *19 (C.D. Cal. Nov. 8, 1999) (rejecting defendants' contention that "plaintiffs' 'work' is the entire daily newspaper because their copyright registration covers the paper as a whole rather than any particular article").

The cases cited above show courts follow a practical, case-by-case approach in determining what should be considered a separate work. Here, the API packages can be considered separate works. Dr. Reinhold described the process by which API packages are developed and added to J2SE through the Java Community Process. He testified, for example, that he submitted a Java Specification Request (JSR) for the java.nio package to the JCP, formed an expert group, and went through 30 drafts over the course of two years before finalizing the java.nio API specification for formal approval. (RT at 624:3-627:17 (Reinhold).) Individual API packages have been separately authored, developed and added by this process for many years, and the number of API packages in Java has increased dramatically over time. Dr. Reinhold

testified that Java 1.0 had seven API packages, Java SE 5 had 166, and Java 7 has 209. (*Id.* at 631:19-25.) He also testified that others created individual API packages that compete with the Java API packages, and used java.util.logging as an example. (*Id.* at 630:11-631:18.) Accordingly, the specifications for the API packages are recognizable works, as are the files for the implementations of the API packages. As in *Texaco*, it was unnecessary for Sun to register separately each part of the APIs, class library source code, class libraries, and compiler and other tools for a given version of the Java platform. *Texaco*, 802 F. Supp at 17. The Copyright Office does not want this either, and its rules do not require it. The separate creations in J2SE are the copyrighted works at issue, not the entire platform. That Google copied from only a subset of the API packages and did not need to copy the remainder is further evidence that the packages are separable works.

C. What will we tell the jury they should be comparing for copyright infringement purposes?

For infringement purposes, the jury should be told they should compare the 37 API packages in Java SE to the corresponding 37 API packages in Android, at the documentation and code levels. Because Google has stipulated, however, that "Android and Java 2 SE Version 5.0 have substantially the same selection, arrangement and structure of API elements," (ECF No. 946), this issue should not have to go the jury.

Similarly, the jury should be told to compare the 12 code files in Android to their corresponding code files in Java SE because the Java SE files are each independent, original computer programs.

Google argues that the portions of Java it copied for Android comprise a relatively small portion of Android as a whole. But it is of no moment that Google added its own work to the 37 APIs copied from Oracle, and the jury should be instructed accordingly. "No plagiarist can excuse the wrong by showing how much of his work he did not pirate." *Shaw v. Lindheim*, 919 F.2d 1353, 1362 (9th Cir 1990) (quoting 4 Nimmer on Copyright § 13.03[B][1][a])); *Sheldon v. MGM*, 81 F.2d 49, 56 (2d Cir. 1936)(J. Hand)(stating the same). Further the Ninth Circuit has consistently held that "a copyright defendant need not copy a plaintiff's work in its entirety to

infringe the work. It is enough that the defendant appropriated a substantial portion of the plaintiff's work. *L.A. Printex Indus., Inc. v. Aeorpostale, Inc.*, No. 10-56187, 2012 U.S. App. LEXIS 7079, at *22-23 (9th Cir. Apr. 9, 2012).

Google's argument that the portion of Java it stole was a relatively small portion of the overall platform is also contrary to Supreme Court law, which emphasizes the need to look at the "qualitative" aspect of what was used. *Harper*, 471 U.S. at 545 (copying of 300 words of quotation from unauthorized manuscript of Gerald Ford memoir was not fair use). In fact, courts have repeatedly held that very small snippets of larger works taken from larger copyrighted works can constitute copyright infringement. *See e.g., Baxter v. MCA, Inc.*, 812 F.2d 421, 425 (9th Cir. 1987) (stating that a six-note sequence taken from a copyrighted song could constitute result in copyright infringement of the larger work); *Fred Fisher, Inc. v. Dillingham*, 298 F. 145 (S.D.N.Y.1924) (L. Hand, J.) (eight note "ostinato" held to infringe copyright in song).

III. COURT'S QUESTIONS FROM ECF 948

A. What case law or other authority is there that states the judge must identify the "work as a whole" (for similarity, fair use, and de minimis) for the jury? Which party has the burden to identify the "entire work"?

Oracle has not found any case law or other authority directly analyzing whether the judge or the jury should make the "work as a whole" determination. Several decisions support that the "work as a whole" determination is a question of law for the Court.

Summary judgment rulings on fair use defenses are the most relevant authority. Fair use is a "mixed question of fact and law." *See, e.g., Hustler*, 796 F.2d at 1150-51. Yet, summary judgment is frequently granted even in the presence of a dispute about the scope of the "work as a whole." In *Super Future Equities, Inc. v. Wells Fargo Bank Minn.*, the court granted summary judgment of copyright infringement, holding that the one copied page from a website was the "whole of the copyrighted work. . . not the entire website." 553 F. Supp. 2d 680, 699-700 (N.D. Tex. 2008) (citing *L.A. Times*, 1999 WL 33644483). Likewise, in *Hustler*, the Ninth Circuit affirmed the lower court's grant of summary judgment of fair use. In reaching its determination, the Ninth Circuit held that the "entire work" was the advertisement that had been copied and not

the whole magazine, even though it made up less than one percent of the 154 page-long magazine. *Id.*

Asking the Court to identify the work as a whole is consistent with the law requiring the court to determine copyrightability and thereby to identify the scope of copyright protection and the relevant standard for comparison for the jury. *See, e.g., Apple Computer, Inc. v. Microsoft Corp.*, 35 F.3d 1435, 1443 (9th Cir. 1994). The court sets the framework for the analysis, and the jury conducts the comparison.

It is particularly appropriate for the Court to decide the issue of the work as a whole here, because Google is claiming that the work as a whole should be the entire J2SE 5.0 platform based on Oracle's pleading and the fact that the platform was the subject of the copyright registration. Both arguments are incorrect, but raise legal issues for resolution by the Court.

With respect to which party bears the burden of identifying the "entire work" in the context of substantial similarity, fair use, or the *de minimis* defense, Oracle has located no case that directly addresses the issue. Logically, the burden should be borne by the party who raises the issue in the course of proving its claims or defenses. As discussed in section III F. below, because of the evidence of Google's direct copying, Oracle need not prove substantial similarity. *Range Road Music, Inc. v. East Coast Food, Inc.*, 2012 U.S. App. LEXIS 3173, at *10 (9th Cir. Feb. 16, 2012). In this case, therefore, Google should bear the burden of identifying the "entire work" because it arises solely in the context of its affirmative defenses. Google bears the burden of proof on fair use. *See Perfect 10, Inc. v. Amazon.com, Inc.*, 508 F.3d 1146, 1158 (9th Cir. 2007). And Google bears the burden of showing that its copying was *de minimis*. *See Merch. Transaction Sys., Inc. v. Nelcela, Inc.*, 2009 U.S. Dist. LEXIS 25663, at *61 (D. Ariz. Mar. 17, 2009) ("Thus, Nelcela will not escape liability unless it can show that the protectable elements in the Lexcel software constitute an insignificant (quantitatively and qualitatively) portion or aspect of the Lexcel software.").

B. With respect to what segment of the "work" can stand alone within the meaning of *Hustler v. Moral*, 796 F.2d 1148, 1155 (9th Cir. 1986), the Court wishes to know whether the implementation of the 37 API packages inherit, call upon, invoke, or incorporate any method, field, or class outside the 37.

Yes. Oracle's implementation of the 37 API packages calls upon and invokes methods, fields, and classes outside the 37, although they do not inherit from or incorporate from outside the 37. The declarations in Oracle's implementation contain relatively few cross-references to API elements outside the 37 packages, and those cross-references are limited to only a few packages, as explained below in Section IV.C.

That there are calls to packages outside the 37, however, should not affect the "work" determination. By analogy, a volume of *Deering's California Code Annotated* may cross-reference and point the reader to other volumes, but that volume alone could still be considered a "work." Similarly, a web page might provide hyperlinks to other web pages, but that should not affect whether the web page could serve as a "work" for infringement purposes.

C. Why shouldn't we let the jury decide what the "work as a whole" is?

As discussed in Section III.A above, the jury should not be permitted to decide what the "work as a whole" is. Google is raising challenges based on legal grounds that are for the Court to decide. In addition, Oracle believes the Court is in a much better position to apply copyright law to determine the appropriate work. The Court should set the framework for the analysis and the jury can then apply it in making a comparison.

D. Is the "work as a whole" the same for purposes of "substantial similarity (or virtually identical)," "fair use," and "*de minimis*" copying? If not, how are the "works as a whole" to be found for these purposes?

Yes. Oracle is aware of no authority and no reason to apply different definitions of the "work as a whole" when analyzing infringement, fair use, or *de minimis* copying.

Of critical importance, however, courts evaluating substantial similarity, fair use and *de minimis* copying require defendants to prove that their copying was both quantitatively and qualitatively insignificant to escape infringement. Google will not be able to meet that standard here as to the APIs for the 37 packages. The evidence shows they were the product of many years of work, (RT at 624:2-626: 15 (Reinhold) (two years for java.nio Package alone)), and

Google is claiming that they were so important it was necessary for Google to copy them. "[E]ven a quantitatively small amount of copied material may be sufficiently important to the operation of a plaintiff's program to justify a finding of substantial similarity. For instance, a small portion of the structure or code of a program may nonetheless give it distinctive features or may make the program especially creative or desirable. In such a case, a finding of substantial similarity is appropriate." *Nimmer on Copyright* § 13.03[F][5] (internal citations omitted). Here Google deliberately engaged in extensive copying of material everyone acknowledges was significant. As the cases cited in section II.C show, Google cannot justify its detailed copying by trying to place it within a larger frame of reference, whether it be under the rubric of substantial similarity, fair use, or *de minimis* copying.

E. For purposes of identifying the "work as a whole," should Oracle be held to the copyrighted work identified in the operative complaint?

No, as a legal matter, but it does not matter because as discussed in section II.B. above, Oracle identified the works in the Amended Complaint. Oracle alleged that Google infringed its copyrights in the API packages, class libraries, and related documentation, and literally copied individual code files, and attached specific examples. (ECF No. 36 ¶ 40 and Ex. I-J). For the API packages, the Amended Complaint specifically referred to "structure and organization" as well as various API elements such as methods and classes that had been copied. (ECF No. 36 ¶ 40).

We have located no decision that requires precise pleading of the "work as a whole" that ultimately will be subject to infringement analysis at trial. As a legal matter, Oracle's only pleading requirement was to give Google fair notice of the claims against it. *See e.g., Home & Nature, Inc. v. Sherman Specialty Co.*, 322 F. Supp. 2d 260, 266-267 (E.D.N.Y. 2004); *Dr. Seuss Enters., L.P. v. Penguin Book USA, Inc.*, 924 F. Supp. 1559, 1563 note 3 (S.D. Cal. 1996). Indeed, Oracle's pleading is more detailed than pleadings deemed sufficient in *Perfect 10 v. Cybernet Ventures*. In *Perfect 10*, the plaintiff's complaint broadly identified "copyrights involving their magazines and identifie[d] ownership of the pictures within the magazines." 167 F. Supp. 2d 1114, 1121 (C.D. Cal. 2001). But the plaintiff did not systematically identify the

precise copyrighted material defendant infringed. The court nevertheless held that this generalized allegation was "sufficient to notify [the Defendant] as to the type of infringing conduct and the source of the claims." *Id.* (denying defendant's motion to dismiss for lack of subject matter jurisdiction). The court reminded defendants that the details of plaintiff's copyrights could be "elicited during the discovery stage." *Id.*

Google may use this or another of the Court's questions to raise a different issue: whether Oracle should be limited to the registrations that were attached to the complaint. To date, Google has not specified how this might affect the claims here. No case throws a plaintiff's copyright claim out because the defendant proves it copied registered Work B while the registration for Work A was the one attached to the complaint.

But even if Oracle were held to the registrations it attached, by referencing copyright registrations that cover works derived from earlier copyrighted works, Oracle properly asserted claims relating to the entire series of copyrighted works. When a copyright owner obtains a registration for a final version in a series of versions of a work, that final registration covers preceding versions of the derivative work. *AFL Telecomms. LLC v. SurplusEQ.com, Inc.*, 2012 U.S. Dist. LEXIS 49239, at *6 (D. Ariz. Apr. 9, 2012). In *AFL*, the court held that in such a case, "it is reasonable to infer…that each successive version incorporates the preceding versions." (*Id.*) Therefore, registration of a derivative work "permits legal actions on preceding versions of the work." *Id.*, at *6; *see also Streetwise Maps v. VanDam, Inc.*, 159 F.3d 739, 747 (2d Cir. 1998) ("the registration certificate relating to the derivative work . . . will suffice to permit [the plaintiff] to maintain an action for infringement based on defendants' infringement of the pre-existing work"); *Salestraq Am., LLC v. Zyskowski*, 635 F. Supp. 2d 1178, 1181 (D. Nev. 2009).

The copyright registrations identified in Exhibit H to Oracle's complaint cover works derived from previous versions of Java. Accordingly, by referencing those registrations, Oracle has asserted registration of its copyrights in both the new material in those registered versions of Java and the older material from which it is derived. In addition, all of the registrations were identified and produced in discovery at the outset of the case and have now been admitted into evidence at trial.

F. If the SSO and declarations are held to be protected elements, then why are there still issues of access and similarity for purposes of infringement (excluding de minimis and fair use)? Put another way, isn't substantial similarity only an issue if there isn't an admission of factual copying of protectable elements?

Substantial similarity is not an issue when there has been an admission of factual copying, as is the case here. The Ninth Circuit recently confirmed this in *Range Road*:

> ***"Substantial similarity" is not an element of a claim of copyright infringement.*** Rather, it is a doctrine that helps courts adjudicate whether copying of the "constituent elements of the work that are original" actually occurred when an allegedly infringing work appropriates elements of an original without reproducing it *in toto*. *See Funky Films*, 462 F.3d at 1076. A showing of "substantial similarity" is irrelevant in a case like this one, in which the Music Companies introduced evidence that the public performances entailed direct copying of copyrighted works. *See id.* (noting that a demonstration of substantial similarity is only necessary to prove infringement, "[a]bsent evidence of direct copying").

Range Road Music, 2012 U.S. App. LEXIS 3173, at *10 (emphasis added). As noted above, Google has admitted copying the 37 API packages and the rangeCheck() file, so a showing of substantial similarity as part of an inferential record of copying is not required.

IV. COURT'S QUESTIONS FROM ECF 951

A. Do any of the Sun compiled lines in the 37 APIs call upon part or all of another API as a step?

Yes. Sun/Oracle's class, method, and field declarations in one package often reference classes defined in another package. An example discussed at the pretrial conference was the class SSLPermission, which is defined in the package javax.net.SSL, and which is a subclass of a class called BasicPermission, which is defined in the java.security package. (ECF No. 895, 3/28/11 Hr'g Tr. at 59:22-60:8.) Other illustrative examples include:

1. The class java.awt.font.NumericShaper has a method called "toString()," which returns a java.lang.String.

2. The class java.net.Socket has a method called "getInputStream()," which returns a java.io.InputStream and throws a java.io.IOException.

3. The class java.nio.channels.Channels has a method called "newWriter," which takes a java.lang.String parameter (among others) and returns a java.io.Writer.

These and other cross-package examples appear consistently in both the J2SE API specifications and in the source code for the class libraries, which in turn is reflected in the object code.

B. If so, do the accused APIs likewise call upon the same other API? That is, even though the implementations are different from the Sun implementation, do the accused APIs borrow from other APIs in the same pattern?

Yes. Google has admitted that Android and J2SE Version 5.0 have substantially the same selection, arrangement and structure of API elements for the 37 packages. Testimony established that Google's overall goal was to copy the pattern and not deviate from it:

> Q. And in your work with these APIs that were already in Android where you were working on the code and trying to improve it and fix the bugs, as to any of them, did you ever make any changes to any of the method declarations or the other elements of the APIs that you identified in the chart that you drew this morning?
>
> A. None whatsoever. I couldn't. It wouldn't have been possible.

(RT at 810:25-811:6 (Bloch).)

C. If the answer to question A is yes, do any of the compiled lines in the 37 Sun APIs call upon part or all of another API outside the 37? Put differently, is all of the cross referencing done within the 37?

Yes, but not very many. Of the classes from the 37 packages copied in Android, only six packages contain declarations that refer to API elements from a package outside the 37. The six packages are java.security.cert, java.security.interfaces, java.security.spec, java.sql, java.util, and javax.sql, and the references outside the 37 are to one package: java.math. The classes in the six packages contain 104 methods or constructors that refer to only two classes in the java.math package: BigDecimal and BigInteger. Other than that, there is no cross-referencing from within the 37 APIs Google copied into Android to outside the 37 APIs. The APIs that Android copied are well-contained.

V. COURT'S QUESTIONS FROM ECF 953

A. What efficiencies, if any, are obtained by grouping methods or fields together under a single class? Put differently, what would be lost if a method that returned the cosine of an angle was grouped under a class other than Math? This discussion should get the pros and cons of the particular interrelationships (e.g., inheritance) within the 37 API packages.

The efficiencies to be gained from the particular relationships within the 37 API packages are primarily efficiencies of software development—specifically, reduced time-to-market and reduced development cost for both platform developers and application developers. A class library that is well-organized is one that is easier to learn, easier to use, and easier to maintain.

Good API designs can also result in performance efficiencies. (RT at 513:9, 516:12, 619:24-621:6, 633:15-634:25, 741:3-742:3, 748:17-22; TX 624 at 20-21.)

Placing methods in specific classes, or classes in specific packages, is important for organizational clarity for users of the library. It is also important for the developers and maintainers of the library. One reason is that the structure and organization provides encapsulation, also called information-hiding or modularity, which is a goal of API design. (TX 624 at 17, 26.)

Class inheritance and Interface implementation is a hierarchical form of organization that provides what is called "subtype polymorphism" and permits code reuse. From the point of view of implementation and maintenance (both improvements and bug fixes), the inheritance hierarchy reduces the number of times an algorithm must appear in the code. A method that is inherited by subclasses need only appear once, in the source code of the parent class. Likewise, a method that operates on objects of a class will also operate on that class's subclasses, so need only appear once. In these cases, the code implementing the algorithm is easier to maintain by the platform developer, because it avoids multiple versions that need to be checked for consistency. Moreover, it is easier for an application developer—the user of the class libraries—to learn only one version of a method, rather than multiple descriptions in multiple class specifications.

Just as importantly, if the cos() method were grouped in a class other than Math, it would diminish the explanatory power of the organization and the ease of use of the class libraries, which arise from the aesthetics of the design of their APIs:

> Q. There are aesthetic matters in API design; correct, sir?
>
> A. Yes, there are.
>
> Q. And it's -- it's not being prissy to think about aesthetic matters. The aesthetics of an API design are part of this noble and rewarding craft. Correct?
>
> A. Yes. Generally, an API that displays good aesthetics will be easy to use. It's like a car's dashboard. Making it pretty isn't just about making it nice to look at. The car will be actually easier to drive if you can see the speedometer.

(RT at 752:5-14 (Bloch).)

Dated: April 22, 2012

MORRISON & FOERSTER LLP

By: /s/ Michael A. Jacobs

Attorneys for Plaintiff
ORACLE AMERICA, INC.