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 MARCH 9, 2012 BRIEF REGARDING COPYRIGHT ISSUES** Dept.: Courtroom 8, 19th Floor Judge: Honorable William H. Alsup

TABLE OF CONTENTS

Page

I. COPYRIGHTS IN SUIT 1
- A. Google copied 37 Java API specifications nearly verbatim into Android 1
- B. Google copied source code, object code and comments from 11 Java code files into Android 3

II. ORACLE'S COPYRIGHT LIABILITY CLAIMS 3
- A. Direct Infringement 3
  - 1. Ownership 4
  - 2. Validity of Copyright 4
  - 3. Copying 6
    - a. Google copied the Java API specifications 6
    - b. Google copied Java Software Code 7
  - 4. Originality 8
- B. Indirect Infringement 9

III. GOOGLE'S AFFIRMATIVE DEFENSES 10
- A. Elements Are Not Protected By Copyright 10
  - 1. Merger and Scenes A Faire 10
  - 2. Other challenges based on functionality 11
- B. Fair Use 12
- C. De Minimis Copying 13
- D. Equitable Defenses 14

IV. EFFECT OF PROGRAMMING LANGUAGE 14

TABLE OF AUTHORITIES

Page(s)

CASES

Bibbero Sys., Inc. v. Colwell Sys., Inc.,
893 F.2d 1104 (9th Cir. 1990) 4, 11

Brocade Commc'ns Sys., Inc. v. A10 Networks, Inc.,
2011 U.S. Dist. LEXIS 91384 (N.D. Cal. Aug. 16, 2011) 6, 8

CDN Inc. v. Kapes
197 F.3d 1256 (9th Cir. 1999) 9

Del Madera Props. v. Rhodes & Gardner, Inc.,
820 F.2d 973 (9th Cir. 1987) 4

Dream Games of Ariz., Inc. v. PC Onsite,
561 F.3d 983 (9th Cir. 2009) 4

Ets-Hokin v. Skyy Spirits, Inc.,
225 F.3d 1068 (9th Cir. 2000) 5, 8, 10, 11

Feist Publ'ns, Inc. v. Rural Tel. Serv. Co.,
499 U.S. 340 (1991) 9

Funky Films, Inc. v. Time Warner Entm't Co.,
462 F.3d 1072 (9th Cir. 2006) 3

Harper & Row Publishers, Inc. v. Nation Enters.,
471 U.S. 539 (1985) 12, 13

Jacobsen v. Katzer,
2009 U.S. Dist. LEXIS 115204 (N.D. Cal. Dec. 10, 2009) 8

Jartech, Inc. v. Clancy,
666 F.2d 403 (9th Cir. 1982) 12

Johnson Controls, Inc. v. Phoenix Control Sys., Inc.,
886 F.2d 1173 (9th Cir. 1989) 9, 11

Lanard Toys Ltd. v. Novelty, Inc.
375 Fed. Appx. 705 (9th Cir. 2010) 5, 11

Louis Vuitton Malletier, S.A. v. Akanoc Solutions, Inc.,
658 F.3d 936 (9th Cir. 2011) 9

Maljack Prods., Inc. v. UAV Corp.,
964 F. Supp. 1416 (C.D. Cal. 1997) 5

MDY Indus., LLC v. Blizzard Entm't, Inc.,
2011 U.S. App. LEXIS 3428 (9th Cir. Feb. 17, 2011) 10

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

Newton v. Diamond,
388 F.3d 1189 (9th Cir. 2004)..... 13

Perfect 10, Inc. v. Amazon.com, Inc.,
508 F.3d 1146 (9th Cir. 2007)..... 12

Pye v. Mitchell,
574 F.2d 476 (9th Cir. 1978)..... 7

Range Road Music, Inc. v. East Coast Food, Inc.,
2012 U.S. App. LEXIS 3173 (9th Cir. Feb. 16, 2012) 6, 13

Satava v. Lowry,
323 F.3d 805 (9th Cir. 2003)..... 14

Swirsky v. Carey,
376 F.3d 841 (9th Cir. 2004)..... 8

Three Boys Music Corp. v. Bolton,
212 F.3d 477 (9th Cir. 2000)..... 6

United States v. King Features Entm't, Inc.,
843 F.2d 394 (9th Cir. 1988)..... 14

STATUTES

17 U.S.C.
§ 107(1) 12
§ 410(c) 4

OTHER AUTHORITIES

3-12 *Nimmer on Copyright* § 12.10[B][1] 8

Bjarne Stroustrup, THE C++ PROGRAMMING LANGUAGE § 16.1 (3d ed. 1997) 15

James Gosling et al., JAVA LANGUAGE SPECIFICATION § 4.3.2 (3d ed. 2005) 15

This brief responds to the Court's Feb. 1, 2012 Order Regarding Copyright Issues and its Feb. 29 supplemental order.

I. COPYRIGHTS IN SUIT

The copyrighted works at issue are (a) 37 Java API design specifications and implementations and (b) 11 Java software code files. Google is liable for copyright infringement because Google copied from and prepared derivative works from Oracle's copyrighted materials to create Android specifications and implementations, which it then distributed.

A. Google copied 37 Java API specifications nearly verbatim into Android

The 37 Java API specifications at issue are the blueprints for the design of Java core class libraries. They describe an extraordinarily complex, intricate structure of hierarchy and interdependency, painstakingly designed by Sun engineers over years of effort. Sun carefully maintained the copyrighted status of the works at issue in this lawsuit. Google infringed the copyrighted works in order to attract the 6-million-strong Java developer community, which enabled Google to capture smartphone market share quickly, advertise thousands of applications in a robust Android market, and create a loyal customer base.

Google initially copied from the documentation for Java 2 Standard Edition 5.0 ("J2SE 5.0"), and later copied from the updated J2SE 6.0 documentation. Oracle's copyright in J2SE 5.0 materials was registered with the U.S. Copyright Office under registration numbers TX 6-066-538 and TX 6-143-306. J2SE 5.0 revised and built upon earlier versions of Java.[1]

Each of the 37 Java API specifications Google copied defines a package, and has a unique name, such as "java.nio" or "java.util." (*See* Mitchell Opening Rep., ECF No. 397 ¶ 202 (listing 37 API specifications at issue).) Packages may include classes, interfaces, and subpackages. For each class in a package, the API specification describes the fields and methods within it and

[1] Sun obtained copyright registrations TX0004616088 [1.2 Beta 2]; TX0005359984 [1.2]; TX0005359985 [1.2]; TX0005359986 [1.2]; TX0005359987 [1.2]; TX0005271787 [1.3]; TX0005316757 [1.3]; TX0005316758 [1.3]; TX0005392885 [1.3]; TX0006196514 [1.4]and TX0006848555 [6.0]. Sun also obtained copyright registrations TX0004326014 on a book entitled "The Java Application Programming Interface, Volume 1 Core Packages," which described an early version of the API specifications.

identifies those that are exposed to other classes. (*See, e.g., id.* ¶ 171; Mitchell Opp. Rep., ECF No. 397-2 ¶¶ 20-28.) Methods are described by their "method signatures," which include the method name and the method parameters and their order, as well as the method return type and any exception types thrown by the method. Fields are described by their name and type. Field and method declarations may also include various modifiers that change their behavior, inheritability, and accessibility. The API specification also describes the relationship of each class to other classes and packages of classes. The Java API designers who wrote the specifications decided which packages, classes, Interfaces, methods, and fields to include in the API specifications, as well as the relationships and interdependencies among the thousands of individual elements. (*See id.*) These relationships can be found within a package or may extend across different packages. This is a vast amount of expression: when printed out, Oracle's 37 API specifications are more than 11,000 pages long. (Swoopes Decl., ECF No. 343 ¶ 28.)

Google copied nearly all of that expression verbatim. In the case of the java.util package alone, Google copied 49 classes, 16 interfaces, 20 exceptions, 85 fields, and 369 methods. Google also copied the structural relationships among these elements within java.util, as well as their relationships to the elements of other packages. For example, according to Oracle's java.util API specification, the Hashtable class is a subclass of Dictionary, and implements three interfaces: Serializable, Cloneable, and Map, which are found in the java.lang, java.io, and java.util packages, respectively. Android's Hashtable class is likewise a subclass of Dictionary, and implements the same three interfaces, with the same three names, found in the same three corresponding Android packages. This organization is hardly preordained, but Google copied it, just as it did for all the other elements of java.util that it wanted for Android.

The Java API implementations are analogous to a multi-volume history, such as the Durants' eleven-volume *The Story of Civilization*, and the Java API specifications are like the author's meticulously detailed outline of those works. It would be copyright infringement for a second author to take the outline and publish a second multi-volume work containing identical names for each volume, chapter, and subchapter, identical topics, facts, and primary sources discussed to the same level of detail, appearing in exactly the same order, and accompanied by

identical photographs and drawings, but change the wording of the text. And it would be copyright infringement for the second author to publish an outline of the second multi-volume work that matches the original author's outline almost exactly.

While Google copied far more than the core library names, its copying of the names is actionable as well. While the Court held individual names are not copyrightable, together the names copied in this case—at least 5,500—are copyrightable if they are "numerous enough and their selection and arrangement original enough that their combination constitutes an original work of authorship." *Merch. Transaction Sys., Inc. v. Nelcela, Inc.*, 2009 U.S. Dist. LEXIS 25663, at *46 (D. Ariz. Mar. 17, 2009) (claim based on selection and arrangement of field names in software). The Court has held that Oracle may present at trial a theory of copyright infringement based on the selection and arrangement of names. (*See* ECF No. 433 at 8.)

B. Google copied source code, object code and comments from 11 Java code files into Android

Google also copied 11 of Oracle's Java software code files into 12 Android code files:

- the entire code for AclEntryImpl.java, AclImpl.java, GroupImpl.java, OwnerImpl.java, PermissionImpl.java, PrincipalImpl.java, PolicyNodeImpl.java, and AclEnumerator.java, obtained by decompiling object code for these classes
- code from Arrays.java
- comments from CodeSource.java
- comments from CollectionCertStoreParameters.java

II. ORACLE'S COPYRIGHT LIABILITY CLAIMS

Oracle has asserted two copyright liability theories: (1) direct infringement and (2) indirect infringement, including vicarious and contributory infringement.

A. Direct Infringement

"A plaintiff bringing a claim for copyright infringement must demonstrate '(1) ownership of a valid copyright, and (2) copying of constituent elements of the work that are original.'" *Funky Films, Inc. v. Time Warner Entm't Co.*, 462 F.3d 1072, 1076 (9th Cir. 2006) (quoting *Feist*

Publ'ns, Inc. v. Rural Tel. Serv. Co., 499 U.S. 340, 361 (1991)). The burdens of proof for these elements are discussed separately below.

1. Ownership

"A certificate of registration raises the presumption of copyright validity and ownership." *Dream Games of Ariz., Inc. v. PC Onsite*, 561 F.3d 983, 987 n.2 (9th Cir. 2009). As noted above, Oracle's copyrights in J2SE 5.0 materials were registered with the U.S. Copyright Office under registration numbers TX 6-066-538 and TX 6-143-306. J2SE 5.0 was registered as a collective work, comprising prior works by Sun, licensed-in components, and new and revised computer code and accompanying documentation and manuals. (*See* ECF No. 36-8 at 7.) The registration form lists Sun's copyright registrations for earlier versions. (*See id.* at 7-8.) Oracle America, Inc. is the same corporate entity as Sun, which changed its corporate name after Oracle acquired it. Thus Oracle America, Inc. is the owner of the registered copyrights in J2SE 5.0 and Sun's other registered copyrights covering earlier and later versions of the Java API specifications, implementations, and other documentation. (*See* n.1, *supra*.)

If Google disputes ownership based on the legal interpretation of copyright regulations or registrations, that interpretation is for the Court to make. If Google is raising a factual challenge to ownership, such as to authorship, that issue is for the jury to decide. *See, e.g., Del Madera Props. v. Rhodes & Gardner, Inc.*, 820 F.2d 973, 980 (9th Cir. 1987) ("The question of authorship of a copyrighted work is a question of fact for the jury.").

2. Validity of Copyright

The parties agree that the Court decides copyrightability. (ECF No. 525 at 10.)

A "certificate of copyright registration constitutes prima facie evidence of copyrightability and shifts the burden to the defendant to demonstrate why the copyright is not valid." *Bibbero Sys., Inc. v. Colwell Sys., Inc.*, 893 F.2d 1104, 1106 (9th Cir. 1990); *see also* 17 U.S.C. § 410(c) ("In any judicial proceedings the certificate of a registration made before or within five years after first publication of the work shall constitute prima facie evidence of the validity of the copyright and of the facts stated in the certificate.").

The presumption that a work is entitled to copyright protection also extends to constituent elements of the work. The presumption can be rebutted.

In *Lanard Toys Ltd. v. Novelty, Inc.*, the Ninth Circuit approved use of a jury instruction requiring defendants to prove elements were not subject to copyright protection:

> As Nimmer explains, "[U]nlike a patent claim, a claim to copyright is not examined for basic validity before a certificate is issued. ... There is one important respect, however, in which the Copyright Office does perform a substantive examination within its particular expertise to determine validity: whether the work falls within the subject matter of copyright. The Office will register only works as to which, 'after examination,' it determines that 'the material deposited constitutes copyrightable subject matter...'" 3-12 Nimmer § 12.11[B][3] (citing 17 U.S.C. § 410(a)). Accordingly, *the district court did not err by instructing the jury that appellants bore the burden of proving that any functional elements of Lanard's toys were not subject to copyright protection*, and that any similarity between their products and Lanard's works were limited to such functional elements.

375 Fed. Appx. 705, 711 (9th Cir. 2010) (omission in original and emphasis added); *see also Maljack Prods., Inc. v. UAV Corp.*, 964 F. Supp. 1416, 1426 (C.D. Cal. 1997) ("The burden of proving invalidity lies with the party making the invalidity claim").

Other courts, however, have characterized the burden as one to come forward with evidence: "'To rebut the presumption, an infringement defendant must simply offer some evidence or proof to dispute or deny the plaintiff's prima facie case of infringement.'" *Ets-Hokin v. Skyy Spirits, Inc.*, 225 F.3d 1068, 1076 (9th Cir. 2000) (citation omitted). *Ets-Hokin* sums up the burden with respect to a photograph this way: "In short, to overcome the presumption of validity, defendants must demonstrate why the photographs are not copyrightable." *Id.*

The Court should follow *Lanard Toys* and hold that Google has the burden of proving that the elements it copied are not subject to copyright protection. But regardless of how it is articulated, Google's burden extends to each feature Google copied. As Judge Murguia held in *Merchant Transaction Systems v. Nelcela*:

> Th[e] presumption extends to the Court's use of analytic dissection under the extrinsic test to determine whether any of the allegedly similar features are protected by copyright. *See Apple Computer*, 35 F.3d at 1443. Because Lexcel satisfied its prima facie burden and was determined to possess a valid copyright, Lexcel is entitled to a presumption that the common features and elements in the Lexcel and Nelcela software are copyrightable. *See Swirsky,* 376 F.3d at 851. As such, contrary to Nelcela's assertion that "it is Lexcel's burden to prove the significance of the elements copied" (Dkt. # 603, p.4), the burden in fact shifts to

> Nelcela to demonstrate why those common features and elements are not protected by copyright.

2009 U.S. Dist. LEXIS 25663, at *26-27 (citing *Bibbero*, 893 F.2d at 1106); *see also Brocade Commc'ns Sys., Inc. v. A10 Networks, Inc.*, 2011 U.S. Dist. LEXIS 91384, at *7 (N.D. Cal. Aug. 16, 2011) ("[I]t is also true that if Brocade registered its software within 5 years of publication, this entitles Brocade to a presumption of validity, and shifts the burden to A10 to rebut that the allegedly copied elements are not protectable expression.") (citing *Merch. Transaction*).

3. Copying

Copying is an issue of fact for the jury, and Oracle bears the burden of proof. *See Three Boys Music Corp. v. Bolton*, 212 F.3d 477, 481-82 (9th Cir. 2000). Oracle can fulfill this burden with evidence of direct copying or by showing that Google had access to the copyrighted work and that there is a substantial similarity between the copyrighted and infringing works. *Id.* at 481. The Ninth Circuit has held there is no need to prove similarity if direct copying is shown:

> "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.*

Range Road Music, Inc. v. East Coast Food, Inc., 2012 U.S. App. LEXIS 3173, at *10 (9th Cir. Feb. 16, 2012). Here, there is substantial evidence proving copying under either theory.

a. Google copied the Java API specifications

Google copied the designs expressed in the Java API specifications, which include thousands of elements and the relationships among them, essentially in their entirety. Google does not deny that this copying took place. To the contrary, its expert, Owen Astrachan, acknowledges that Google copied the API specifications deliberately, opining that "Even the slightest changes to the names or organization of API elements will thwart compatibility and interoperability, because existing code that used those elements would not run properly, and programmers would have to learn new API element names." (Astrachan Opening Rep., ECF No. 262-1 ¶ 131). This is sufficient to satisfy Oracle's burden to show copying under *Range Road Music, supra.* It is no defense for Google to claim it copied from a third party. A defendant is

liable when it copies from a third party, with or without a license. *Pye v. Mitchell*, 574 F.2d 476, 481 (9th Cir. 1978).

While Oracle is not required to also prove copying through access and substantial similarity, it can do so. The Java API specifications were available on the Sun (now Oracle) website, subject to explicit copyright protection, and Google admits Android developers had access to them. (Google Resp. to RFA No. 168.) Bob Lee, the lead developer for the Android core libraries, testified that Google worked to "implement core libraries *according to Java APIs*." (Lee Dep. 14:7-11 (emphasis added).) Mr. Lee "consulted Sun's website for the API specifications when doing the work for Google." (*Id.* at 66:7-9.) He did so knowing that the specifications were subject to a copyright notice. (*Id.* at 65:8-67:5.)

There is no real question of substantial similarity. (*See, e.g.,* Mitchell Opening Rep. Ex. Copyright A-D, ECF No. 397 at 124-160 (comparing specifications).) On summary judgment, the Court ruled as to the API specifications that "Because Google has not proven that a substantial portion of the specifications is unprotected, Google's justification for applying the virtual identity standard fails." (ECF No. 433 at 11.) Google will not be able to prove so at trial either. However, Oracle could satisfy a virtual identity standard as well.

b. Google copied Java Software Code

Oracle will show that Google copied source code, object code, and comments from 11 Java software code files. Google's expert has not denied this copying, but only denies its significance. (*See* Astrachan Opening Rep., ECF No. 262-1 ¶¶ 145-175.) Again, evidence of this direct copying satisfies *Range Road Music*, but Oracle can show access and similarity as well. Google engineer Josh Bloch, formerly of Sun, conceded that there "is a strong indication that it is likely" that he accessed the Oracle source code for the method called "rangeCheck" when he wrote the same method in Android. (Bloch Dep. 181:7-14.) And the evidence shows Google decompiled Java object code, which can be retrieved from most desktops or many other places Java is available. A side-by-side comparison of the copied code and comments shows they are virtually identical. (*See* Mitchell Opening Rep., ECF No. 397 ¶¶ 77, 206-26.)

4. Originality

Google's challenge to originality is a question for the Court to decide. *See* 3-12 *Nimmer on Copyright* § 12.10[B][1]. But threshold determinations of fact are for the jury. *Id.* When there are no genuine issues of fact, as is the case here, originality may be resolved as a matter of law. *Jacobsen v. Katzer*, 2009 U.S. Dist. LEXIS 115204, at *9-10 (N.D. Cal. Dec. 10, 2009).

The Court denied Oracle's request to file a summary judgment motion on the issue of originality of selection and arrangement of names of APIs, stating that "The judge will understand the complications of the issues much better after the detailed trial evidence is laid out and cross-examined. It would be hard for the judge to master the material with the same grasp via a cold written record." (ECF No. 584.) Accordingly, the Court should decide the originality issue on a full record, after hearing fact and expert witness testimony regarding the API design, and should instruct the jury on this issue. While the Court may determine it wishes to submit a threshold issue of fact to the jury, Oracle does not believe that will be necessary.

Because its copyrights were timely registered, Oracle is entitled to a presumption of originality as to the copied elements, and the burden shifts to Google to demonstrate why they are not original. *See, e.g., Ets-Hokin*, 225 F.3d at 1075-76 ("In short, to overcome the presumption of validity, defendants must demonstrate why the photographs are not copyrightable."). Moreover, a copyright registration creates a presumption of originality as to the *specific elements* of a registered work. *See, e.g., Swirsky v. Carey*, 376 F.3d 841, 851 (9th Cir. 2004) (valid registration certificate entitled plaintiff to presumption of originality "as to first measure of chorus" of song); *Merch. Transaction*, 2009 U.S. Dist. LEXIS 25663, at *26-27 (registration shifts burden to defendant "to demonstrate why those common features and elements are not protected by copyright"); *Brocade*, 2011 U.S. Dist. LEXIS 91384, at *7 (registration shifts burden to defendant "to rebut that the allegedly copied elements are not protectable expression"). Google's burden is a difficult one to meet, in part because there is only a minimal degree of originality required. In ruling on summary judgment on the copyrightability of individual names, the Court stated that Oracle had not shown that it was entitled to a presumption of originality because it cited "no authority requiring a presumption of *originality* as to *specific elements* of a registered

work." (ECF No. 433 at 8 (emphasis in original).) Based on the authority cited immediately above, however, the presumption should be applied at trial.

Regardless, the legal threshold for originality is low: "To be sure, the requisite level of creativity is extremely low; even a slight amount will suffice." *Feist*, 499 U.S. at 345 (citation omitted). In *CDN Inc. v. Kapes*, for example, the Ninth Circuit found that prices in a guide for collectible coins were protected by copyright, emphasizing that the required level of originality is "minimal." 197 F.3d 1256, 1262 (9th Cir. 1999). The court explained that the white pages phone listings in *Feist* "did not qualify because they fell into the 'narrow category of works in which the creative spark is utterly lacking or so trivial as to be virtually nonexistent." *Id.* at 1259-60 (quoting *Feist*, 499 U.S. at 359).

The API specifications express a highly creative, complex interplay of elements. Oracle will present evidence at trial about the many design choices involved in creating the Java APIs, their uniqueness, and the skill required to write them. Google's own employees have extolled the creative expression in a well-constructed API. For example, former Sun engineer and current Google employee Josh Bloch testified that a good API will "read like prose" and wrote that "API design is a noble and rewarding craft." (Bloch Dep. 81:13-82:7, 90:8-91:11.) No court has ever held that a structure with the level of expression, intricacy, complexity and interdependency present here lacks the creativity to qualify for copyright protection.

Google cannot mount a serious challenge to the originality of code it literally copied either. Indeed, its only defense is that its copying was minimal. "Source and object code, the literal components of a program, are consistently held protected by a copyright on the program." *Johnson Controls, Inc. v. Phoenix Control Sys., Inc.*, 886 F.2d 1173, 1175 (9th Cir. 1989).

B. Indirect Infringement

Oracle has claims against Google for both contributory and vicarious infringement.

To establish contributory infringement Oracle must show: (1) Google knew or had reason to know of the infringing activity of others; and (2) Google intentionally or materially contributed to the infringing activity. *Louis Vuitton Malletier, S.A. v. Akanoc Solutions, Inc.*, 658 F.3d 936, 943-44 (9th Cir. 2011). Intent to infringe is not required. *Id.* at 943.

To establish vicarious infringement, Oracle must show that: (1) Google profited directly from the infringing activity of others; (2) Google had the right and ability to supervise and control the infringing activity; and (3) Google failed to exercise that right and ability. *MDY Indus., LLC v. Blizzard Entm't, Inc.*, 2011 U.S. App. LEXIS 3428, at *10 (9th Cir. Feb. 17, 2011).

The evidence will show Google *requires* mobile manufacturers to include the 37 infringing API implementations on their devices as part of its licensing terms, and has profited handsomely from the infringing activity. Google also knowingly included the infringing rangeCheck code on Android handsets, where it remains to this day. Google has the ability to force updates to code on existing handsets remotely, but has not done so. Google published the other 10 files containing copied code and comments on its Android developer website, but took them down after it was caught.

III. GOOGLE'S AFFIRMATIVE DEFENSES

A. Elements Are Not Protected By Copyright

Google's answer incorporates several different arguments under this general heading. (*See* ECF No. 51 at 10-11.)

1. Merger and Scenes A Faire

In the Ninth Circuit, merger and *scenes a faire* are both viewed as defenses to a claim of infringement, rather than challenges to copyrightability. *See Ets-Hokin*, 225 F. 3d at 1082. Google accordingly has the burden of proof. *Merch. Transaction*, 2009 U.S. Dist. LEXIS 25663, at *27-28.

In ruling on summary judgment, the Court rejected Google's attempt to claim that entire categories of elements in the APIs were unprotectable, stating that, "This order declines to hold that API package specifications, or any particular category of elements they contain, are unprotectable under the *scenes a faire* or merger doctrines." (ECF No. 433 at 9.) The Court ruled that, "If Google believes, for example, that a particular method declaration is a *scene a faire* or is the only possible way to express a given function, then Google should provide evidence and argument supporting its views as to that method declaration." (*Id.*)

Merger and *scenes a faire* are issues for the Court to decide. *Ets-Hokin,* 225 F.3d at 1082. But these fact intensive questions will require examination of a full record at trial. The Court may wish to seek an advisory verdict from the jury on the underlying factual question of whether a challenged element is "the only possible way to express a given function" (ECF No. 433 at 9) or a "commonplace expression[] that [is] indispensable and naturally associated with the treatment of a given idea." (*Id.* at 8 (quoting *Swirsky v. Carey,* 376 F.3d at 850).)

2. Other challenges based on functionality

Google generally alleges as its affirmative defense that the copyrights at issue are not protectable because they are works that are "functional." (ECF No. 51 at 10.) It has framed this argument in various ways throughout the case, claiming that the APIs are unprotectable "methods of operation" or "functional requirements for compatibility." The parties agree that copyrightability is an issue of law to be addressed by the Court. (ECF No. 525 at 10.)

The Court should follow the Ninth Circuit's decision in *Lanard Toys* and hold that Google has the burden of proving the allegedly functional elements it copied are not subject to copyright protection. *Lanard Toys*, 375 Fed. Appx. at 711. Oracle's registration certificates create a rebuttable presumption that shifts the burden to Google to demonstrate why Oracle's copyrights as to the API specifications are invalid. *Bibbero*, 893 F.2d at 1106.

Merely labeling something "functional" or a "method of operation" does not decide the issue of copyrightability. All computer programs are functional. "Whether the non-literal components of a program, including the structure, sequence and organization and user interface, are protected depends on whether, on the particular facts of each case, the component in question qualifies as the expression of an idea, or an idea itself." *Johnson Controls*, 886 F.2d at 1175. To establish that an element is disqualified from copyright protection because of functionality, Google should be required to satisfy the merger doctrine's requirement that it is "the only possible way to express a given function." (*See* ECF No. 433 at 9 (citing *Satava v. Lowry*, 323 F.3d 805, 812 n.5 (9th Cir. 2003).)

The Court has already rejected Google's argument that the API specifications are "methods of operation." (*Id.* at 11.) As discussed in Oracle's opposition to summary judgment,

Google's method of operation argument relies on *Lotus v. Borland*, which is factually distinguishable because it related to a much simpler consumer user menu. Just as importantly, *Lotus* has never been followed in the Ninth Circuit, and other Circuit courts have expressly refused to adopt its reasoning. (*See* ECF No. 339 at 9-12.)

Google's tautological claim that it copied "functional requirements for compatibility" — by which it means it copied in order to be compatible with those features of Java it wished to copy — should be evaluated as part of Google's fair use defense, discussed below.

B. Fair Use

The parties agree that the jury should decide Google's fair use defense. (ECF No. 525 at 15.) *See, e.g.*, *Jartech, Inc. v. Clancy*, 666 F.2d 403, 405, 407 (9th Cir. 1982) (upholding special jury verdict finding that defendants' use of films was fair use). Google bears the burden of proving fair use. *Perfect 10, Inc. v. Amazon.com, Inc.*, 508 F.3d 1146, 1158 (9th Cir. 2007).

The Court previously rejected Google's efforts to determine fair use as a matter of law, finding that "fact issues preclude a summary judgment finding." (ECF No. 433 at 13.) Google must therefore present its defense of fair use to the jury. However, Oracle believes a directed verdict in its favor is appropriate because Google cannot meet its burden of proof.

In determining whether a use is fair, 17 U.S.C. § 107 sets out four factors to consider, although other factors may be considered as well:

(1) Purpose and character of use – This expressly includes "whether such use is of a commercial nature or is for nonprofit educational purposes." 17 U.S.C. § 107(1). Google's use is clearly commercial, which "tends to weigh against a finding of fair use." *Harper & Row Publishers, Inc. v. Nation Enters.*, 471 U.S. 539, 562 (1985). The Supreme Court held that "[a]lso relevant to the 'character' of the use is 'the propriety of the defendant's conduct.'" *Id.* (quoting 3 *Nimmer on Copyright* § 13.05[A], at 13-72). Google's decision to push forward with Android without a license, when it knew one was required, weighs strongly against fair use.

(2) Nature of the copyrighted work – The Java APIs are highly creative works, as discussed above. (*See, e.g.*, Astrachan Dep. 128:9-13 ("Just as it's hard to find people that are really good at anything that's hard, whether it be, you know, being an artist, a football player, a

concert violinist. Those things are hard. This [API design] is something that's hard in the same way."); *see also* evidence discussed at section I.A, *supra*.)

(3) Amount and substantiality of the work used – This inquiry includes both qualitative and quantitative evaluations. In *Harper*, the Supreme Court found that copying even 300 words from Gerald Ford's unpublished manuscript was not fair use because of the "qualitative nature of the taking." 471 U.S. at 545, 564-65. Google copied from the core Java API specifications — about one-quarter of the Java APIs in total — and will not be able to establish that its copying was quantitatively or qualitatively insignificant.

(4) Effect of the Use on the Market for or Value of the Copyrighted Work – The Supreme Court has stated that "This last factor is undoubtedly the single most important element of fair use." *Id.* at 566-67 (citation omitted). Google's copying has dramatically impaired the market and value of Java, by creating a free competing platform in the mobile device arena, and by fragmenting Java and undermining its "write once, run anywhere" promise.

C. De Minimis Copying

Google asserts its literal copying of Oracle source code and comments is not actionable because it is *de minimis*. "[A] use is de minimis only if the average audience would not recognize the appropriation." *Newton v. Diamond*, 388 F.3d 1189, 1193 (9th Cir. 2004). Whether the "average audience" would recognize Google's copying is a jury issue. Google has the burden to show its copying was de minimis. *See, e.g., Range Road Music*, 2012 U.S. App. LEXIS 3173, at * 10 (plaintiff with evidence of direct copying not required to show substantial infringement).

The Court, however, should instruct the jury which "works" it should compare. Google copied eight source code files in their entirety, and those are the works the jury should compare. Google argues its code copying should be compared to the entire Java platform because Oracle registered it as a single work. The Court rejected this argument on summary judgment, finding that "Google has not shown that the Java platform is the proper basis for comparison." (ECF No. 433 at 6.) The Court ruled that "[t]he plain meaning of [37 C.F.R. § 202.3(b)(4)(i)(A)] 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." (*Id.*)

But if Google is correct that its copying should be compared to the Java platform as a whole, the significance of its copying cannot be determined by examining its literal code copying in isolation. The jury should also take into account its copying of the 37 API specifications.

D. Equitable Defenses

Finally, Google raises defenses of implied license, equitable estoppel, laches, and waiver. Google has the burden to prove all elements of these defenses. *See, e.g.*, *United States v. King Features Entm't, Inc.*, 843 F.2d 394, 399 (9th Cir. 1988) (affirming summary judgment where defendant "has not shown that he has evidence to establish three of the[] elements" of estoppel.)

The parties agree these are equitable defenses for the Court to decide. Nevertheless, Google has requested an advisory verdict from the jury on these defenses. There is no reason for jury input on these defenses, because they are equitable, meritless, and prejudicial. Google will not be able to establish that there is a triable issue on the necessary elements of these defenses, such as reasonable reliance, and they can be decided against Google as a matter of law.

IV. EFFECT OF PROGRAMMING LANGUAGE

The Court has asked "whether or not the copyrightability of the selection, arrangement, and structure of the APIs depend on the underlying programming language being Java as opposed to Python or QBASIC or other non-Java-programming language." (ECF No. 754.) The Java programming language provides many options that facilitate expressive, copyrightable API design. But it does not compel or require any particular design.

The Java programming language places almost no requirements on the selection, arrangement, and structure of the Java API elements. Java is a powerful, flexible, state-of-the-art programming language for expressing APIs. It includes several features, such as interfaces and packages, intended to facilitate API design. This empowers developers to express Java API specifications in myriad ways, and supports their copyrightability. *Cf. Satava*, 323 F.3d at 812 n.5 ("Under the merger doctrine, courts will not protect a copyrighted work from infringement if the idea underlying the copyrighted work can be expressed in only one way, lest there be a

monopoly on the underlying idea."). But, while the language created the opportunity for Sun to develop rich, expressive APIs, it did not compel Sun to do so. Sun made a deliberate design decision to include extensive class libraries and a robust set of APIs. In contrast, although C++ is as powerful a language as Java, it has a much more Spartan set of APIs, reflecting the minimalist philosophy of their designer. (*See* Bjarne Stroustrup, THE C++ PROGRAMMING LANGUAGE § 16.1 (3d ed. 1997).) Only a very small number of API elements are so closely tied to the language that they are formally required by the language specification. For example, the language requires the classes java.lang.Object, java.lang.Class and java.lang.String to be present. (*See* James Gosling et al., JAVA LANGUAGE SPECIFICATION § 4.3.2 (3d ed. 2005).)

QBASIC is a simpler programming language than Java, with fewer linguistic options. It is not an object-oriented language, and so has fewer alternatives available for expressing arrangement and structure of API elements. It can be compared to a human language that has only the present tense: there are obviously more expressive options in languages that allow for past, future, and other tenses. Python is a more extensive and sophisticated language than QBASIC, but still does not include all of the same features that Java has to facilitate API design. For example, Python's method signatures do not have types associated with the parameters. In Java, an API designer can give each parameter a type, such as a class or interface created by the designer, expressing how developers should use the parameter. This form of expression is unavailable in Python and related languages.

Like Sun, Google had many options in creating the Android API specifications and libraries, as illustrated by the other APIs it chose to design itself without copying from Sun. Google took what it wanted. It copied by choice, not by necessity.

Dated: March 9, 2012

MORRISON & FOERSTER LLP

By: /s/

Michael A. Jacobs
Attorneys for Plaintiff
ORACLE AMERICA, INC.