Oracle Am., Inc. v. Google Inc.
750 F.3d 1339 (Fed. Cir. 2014)

KACEE TAYLOR*

BACKGROUND

The plaintiff is Oracle America, Inc. (Oracle), a computer technology developer. The defendant, Google Inc. (Google), is a multinational corporation specializing in Internet-related products and services.

In 2010, Oracle acquired Sun Microsystems, Inc. (Sun), a company that developed the Java "platform" for computer programming. Java is a programming language used by developers to write code. The Java platform enabled written programs to be run across various types of hardware instead of having to code for multiple individual machines, thus improving speed and efficiency for developers.

Within the Java platform, Sun created 166 "packages" of common, ready-to-use code for developers to use as shortcuts instead of having to write their own code from scratch. The issue at hand involves 37 of these "packages," commonly known as an application program interface (API). Each API consists of two types of source code, declaring and implementing.[1] The declaring code acts as the "header," or label, which helps identify and introduce the proper prewritten name and functionality. The implementing code provides instructions for carrying out the functions, or the written programs.

Through the acquisition of Sun, Oracle became the owner of the copyrights to the Java platform and API packages. Oracle offers different types of licensing options for those wanting to use the packages, one being a Commercial License that requires the licensee's programs to remain compatible with the Java platform. In 2005, Google and Sun began discussing a potential licensing deal where Google would adapt the Java platform for mobile devices. They also discussed the possibility of partnering with Sun to make the technology part of Google's open-source, mobile platform, Android. The companies were unable to reach an agreement because Google refused to meet the compatibility requirements set forth by the license agreement.

Since the parties reached an impasse, Google created its own

---

* Katherine Taylor is a 2017 Juris Doctor candidate at the University of San Francisco School of Law.

1. Oracle Am., Inc. v. Google Inc., 750 F.3d 1339, 1349 (Fed. Cir. 2014).

version of the Java packages to conform to the Android operating system. Google wrote its own implementing code, with the exception of the "rangeCheck" function and eight decompiled security files. Google incorporated Oracle's declaring code verbatim in order for Java developers to easily recognize and find the packages within Android. However, the declaring code consisted of what the district court coined the "'structure, sequence, and organization' or 'SSO' of the 37 packages."[2] The SSO contained what Oracle deemed to be their system of organization, or elaborately organized taxonomy. Oracle claimed that by building their own version of these packages, Google both rendered the Android operating system incompatible with the Java platform, and infringed on Oracle's copyrights associated with the programming language.

PROCEDURAL HISTORY

In 2012, Oracle brought suit against Google for patent and copyright infringement. The district court and jury simultaneously assessed twenty-four witness testimonies. Google admitted to copying the declaring code in the 37 API packages as well as the rangeCheck function and the security files, but asserted that the use of the latter two was de minimis.

On May 7, 2012, the jury delivered a verdict in favor of Oracle for copyright infringement on the 37 API packages and the rangeCheck function. However, the jury did not find infringement as to the eight security files and were hung on the copyright issue of fair use.

Post-trial, Oracle filed a motion for judgment as a matter of law (JMOL) with regard to the eight security files, and Google filed a motion for JMOL for the rangeCheck function. The district court granted JMOL in favor of Oracle for the eight security files on the grounds that Google admitted to copying the files, and that the copying was not de minimis. The court also concluded that Oracle's API packages were not subject to copyright protection under the federal Copyright Act.

Oracle appealed the court's judgment against its claim for copyright infringement, and Google cross-appealed with regard to the rangeCheck function and security files.

ISSUE

The U.S. Court of Appeals for the Federal Circuit addressed the issues of: (1) whether Oracle's 37 API packages should be protected under copyright pursuant to the Copyright Act; and (2) whether Google committed copyright infringement by replicating Oracle's

---

2.   *Id.* at 1351.

organization and taxonomy of the Java platform for the Android operating system.

DECISION

The Federal Circuit reversed the lower court's decision that the 37 API packages were not protected under copyright, and reinstated the jury's infringement verdict. The court remanded the fair use issue, on which the jury deadlocked, and affirmed Oracle's motion for JMOL for the eight security files. The court denied Google's motion for JMOL with respect to the rangeCheck method. In sum, the court affirmed-in-part, reversed-in-part, and remanded the case for further consideration.

REASONING

It was undisputed that Google copied the Java code owned by Oracle. The issue was whether the 37 API packages in question were protected under copyright. When examining the lower court's decision, the court examined (1) what material is protected under copyright laws; and (2) the scope of conduct for copyright infringement.

I. COPYRIGHTABILITY

The Copyright Act[3] protects computer programs as long as the work is original, and the protection only extends to the expression of an idea.[4] The components of a computer program that constitute "an expression of an idea" are the literal and non-literal elements. The literal elements include the source code and object/binary code, and the non-literal components involve the structure, sequence, and organization of the code. Google admitted that it copied the literal elements of the packages—the declaring source code—and therefore the court did not engage in further discussion on that issue.

Regarding protection of the non-literal elements, courts rely on the "abstraction-filtration-comparison" test.[5] This test can be broken down into three parts: (1) determine the core structural parts that have been allegedly infringed upon; (2) determine and remove all the "non-original" material; and (3) assess the remaining material for violation.[6]

The court stated that the district court did not apply this test, but instead relied on other case precedent that contradicted the binding

---

3.   Copyright Act of 1976, Pub. L. 94-553, 90 Stat. 2541 (codified as amended at 17 U.S.C. §§ 101–810 (2012)).

4.   *Oracle Am., Inc.*, 750 F.3d at 1355 ("[C]opyright protection extends only to expression, not to ideas, systems, or processes . . . .").

5.   *Id.* at 1357.

6.   *Id.* at 1357–58.

decisions. The district court labeled the SSO as a "method of operation" (i.e., serving a functional purpose), and therefore held it was not protected under copyright law. The court disagreed and distinguished the current circumstances based on Oracle's SSO not being a "method of operation," stating that some "methods of operation" are eligible for copyright protection.

Applying the abstraction-filtration-comparison test, the court found that the literal and non-literal elements of the Java language were (1) original and creative; and (2) did violate copyright principles because Google could have re-created, designed, and hierarchically structured the 37 API packages differently while achieving the same functions within its Android mobile operating system.

II. SCOPE OF CONDUCT

When determining the scope of conduct for copyright infringement, the fair use doctrine may be used as an affirmative defense under the Copyright Act. Courts evaluate this defense on a case-by-case basis, taking into consideration four non-exclusive factors. These factors are: (1) the purpose and use of the copyrighted work, (2) the nature of the work, (3) the amount and substantiality used proportionate to the work as a whole, and (4) the effect on the market for using the copyrighted work.[7]

In evaluating the first factor, courts look at: (1) whether and how much the new work alters the original creation, and (2) whether the use is for commercial purposes.[8] It was undisputed that Google used the work for commercial purposes, but the court could not conclude if and to what extent the copyrighted work was transformative due to the parties disputing material facts.

The second factor concerns the nature of work in question. Computer programs are by nature functional, but also can be considered expressive—often creating blurry lines in determining fair use. Ultimately, if it is imperative to copy the elements exactly to achieve functionality, an argument supporting fair use exists. The court insinuated support in favor of finding fair use by commenting on the necessity of copying three core API packages (out of the 37) in order for the Java-language programs to properly function.

The third fair use factor focuses on the substantive value of the copyrighted work. The court noted that usually a large portion of work copied verbatim, as in this instance, denotes high worth and substance, thus showing evidence against fair use.

The fourth factor was the most important to the court. This factor takes into account the harm to the overall market that is caused by the infringement, as well as the adverse affect that the defendant's

---

7. 17 U.S.C. § 107 (2012).

8. *Oracle Am., Inc.*, 750 F.3d at 1374.

actions have on the originator's marketability. In the same fashion as the first factor, disputes of material facts prevented the court from concluding the effect on the market.

Overall, the court remanded the question of fair use to the district court based on factual disputes surrounding the first and fourth factors.

Lastly, the court addressed the issues of the rangeCheck function and the eight decompiled security files. Google argued that even though they may have infringed, it was de minimis, and therefore the motion for JMOL should not be granted. The court concluded that Google's replication of the rangeCheck function was not de minimis, and that the district court's decision regarding Google's defense for the security files was correct.

Based on the analysis above, the court granted Oracle's motions for JMOL; remanded the question of fair use to the lower court because of existing disputes of material fact; and reversed the lower court's decision of copyrightability, reinstating the jury's infringement verdict.