

5-2017

Transformative Use in Software

Clark D. Asay

BYU Law, asayc@law.byu.edu

Follow this and additional works at: https://digitalcommons.law.byu.edu/faculty_scholarship

 Part of the [Intellectual Property Law Commons](#)

Recommended Citation

Clark D. Asay, [Transformative Use in Software](#), 70 STAN. L. REV. ONLINE 9 (2017).

This Article is brought to you for free and open access by BYU Law Digital Commons. It has been accepted for inclusion in Faculty Scholarship by an authorized administrator of BYU Law Digital Commons. For more information, please contact hunterlawlibrary@byu.edu.



ESSAY

Transformative Use in Software

Clark D. Asay*

Introduction

Fair use is copyright law's most important defense against claims of copyright infringement.¹ Major corporations depend on it to pursue a variety of technological innovations;² universities rely on it for a number of educational purposes;³ and innovative parties frequently resort to it in creating works that build upon the creativity of others.⁴ In short, fair use is an essential limitation on the rights of copyright holders that helps copyright fulfill its constitutional purpose of "promot[ing] the Progress of Science and useful Arts."⁵

In determining whether Party A's use of Party B's copyrighted work is a fair use, courts primarily consider four statutory factors.⁶ Within that inquiry, whether the use is "transformative" has become one of the most important considerations.⁷ In fact, if a court deems a use to be transformative, the court

* Associate Professor of Law, BYU Law School.

1. MARSHALL A. LEAFFER, UNDERSTANDING COPYRIGHT LAW 495 (6th ed. 2014) (indicating that fair use is "by far the most important defense to an action for copyright infringement").
2. *See, e.g.*, *Sony Corp. of America v. Universal City Studios, Inc.*, 464 U.S. 417, 425, 427, 442 (1984) (finding using VCRs to copy entire copyrighted programs for time-shifting purposes to be fair use).
3. *See, e.g.*, *Cambridge Univ. Press v. Patton*, 769 F.3d 1232, 1237, 1242-43 (11th Cir. 2014) (discussing professors within the Georgia university system relying on fair use to make digital excerpts of copyrighted texts available to their students).
4. *See, e.g.*, *Cariou v. Prince*, 714 F.3d 694, 698 (2d. Cir. 2013) (finding Richard Prince's reuse of copyrighted photographs was fair use in part because he added his own creativity to the photographs).
5. U.S. CONST., art. I, § 8, cl. 8; *see also* *Campbell v. Acuff-Rose Music, Inc.*, 510 U.S. 569, 575 (1994) ("From the infancy of copyright protection, some opportunity for fair use of copyrighted materials has been thought necessary to fulfill copyright's very purpose, '[t]o promote the Progress of Science and useful Arts . . .'" (alteration in original) (quoting U.S. CONST., art. I, § 8, cl. 8)).
6. *See infra* Part I.
7. Matthew Sag, *Predicting Fair Use*, 73 OHIO ST. L.J. 47, 55 (2012).

often either discounts the other factors or finds that the transformative nature of the secondary use heavily influences how to assess them.⁸

In what may prove to be one of the most important software copyright cases in decades,⁹ a jury recently handed Google a victory by concluding that Google's use of some of Oracle's Java software in its Android platform constituted fair use.¹⁰ Oracle has appealed the decision, claiming, among other things, that Google's use of its software cannot be fair use because it does not involve a transformative use.¹¹ This is so, according to Oracle and its amici, because Google uses Oracle's software as software.¹² In other words, because Google uses Oracle's software to perform the same computing functions for which it was originally designed, Google's use cannot be transformative. And if not transformative, the use can hardly be fair.

This Essay contends that accepting Oracle's argument would mean that fair use rarely if ever applies in the software context. Software's functional nature, after all, means that reuses of software will inevitably involve the software carrying out the same functions for which it was designed. But if that reality alone forecloses the possibility of a fair use defense, it means the productive balance that fair use helps strike between copyright holders and follow-on software innovators may be imperiled.

This Essay first provides some background on fair use and the transformative use component of the inquiry. It then lays out more fully Oracle and its amici's argument that uses of software as software are nontransformative and, accordingly, not fair uses. The Essay counters this argument, suggesting that accepting it would largely eliminate fair use as a defense in the software context, thereby threatening robust software innovation. It concludes with some thoughts on how best to apply the transformative use inquiry in software cases.

I. A Fair Use Primer

Copyright's fair use defense excuses activities that would otherwise infringe a party's copyright. It does so in order to "avoid rigid application of the

8. *Id.*

9. See Klint Finley, *The Oracle-Google Case Will Decide the Future of Software*, WIRED (May 23, 2016, 7:00 AM), <https://www.wired.com/2016/05/oracle-google-case-will-decide-future-software>.

10. Jack Nicas, *Google Wins Java Copyright Case Against Oracle*, WALL ST. J. (May 26, 2016, 7:03 PM ET), <https://www.wsj.com/articles/google-wins-java-copyright-case-against-oracle-1464294647>.

11. See *infra* Part II; see also Scott Graham, *5 Takeaways from Oracle's Appeal of Google's Jury Win*, LAW.COM (Feb. 14, 2017), <http://www.law.com/sites/almstaff/2017/02/14/5-takeaways-from-oracles-appeal-of-googles-jury-win-2>.

12. See *infra* notes 35-41 and accompanying text.

copyright statute” in ways that “would stifle the very creativity which that law is designed to foster.”¹³ Hence, fair use helps “ensure that subsequent authors and the public may build upon the work of earlier authors”¹⁴ in cases where doing so is deemed to promote creativity overall without unduly harming the original author’s economic prospects.

The genre of parody, in which a party uses another’s work to comment on or critique it, is a common fair use example.¹⁵ But fair use goes beyond parody. The Copyright Act’s Section 107, which codifies the fair use defense, lists a number of examples that may qualify, such as “criticism, comment, news reporting, teaching . . . , scholarship, or research.”¹⁶ The statute then goes on to specify four factors courts should assess in determining whether any particular use of a copyrighted work constitutes fair use: 1) “the purpose and character of the use,” 2) “the nature of the copyrighted work,” 3) “the amount and substantiality of the portion used in relation to the copyrighted work as a whole,” and 4) “the effect of the use upon the potential market for or value of the copyrighted work.”¹⁷

With regard to the first factor, “the purpose and character of the use,” modern courts frequently focus on whether the use is transformative.¹⁸ If the use is not considered transformative, the defendant faces an uphill battle winning a fair use defense.¹⁹ But if a court determines that the use is transformative, it becomes very likely that the fair use defense will prevail.²⁰

What is a transformative use? In the Supreme Court’s words, it is a use that “adds something new, with a further purpose or different character, altering the [original work] with new expression, meaning, or message.”²¹ Hence, it is not transformative to merely repackage the original work. Instead, the party claiming fair use must have used the work in a different way than the original party. For example, the Second Circuit found that it was transformative to use images originally found on Grateful Dead posters and concert tickets in a coffee-table book.²² The court considered this use transformative because the creator of the book placed the images in a different setting with a different

13. *Campbell v. Acuff-Rose Music, Inc.*, 510 U.S. 569, 577 (1994) (quoting *Stewart v. Abend*, 495 U.S. 207, 236 (1990)).

14. 4 WILLIAM F. PATRY, PATRY ON COPYRIGHT § 10:2, Westlaw (database updated March 2017).

15. *See, e.g., Campbell*, 510 U.S. at 579.

16. 17 U.S.C. § 107 (2015).

17. *Id.*

18. *Sag*, *supra* note 7, at 55.

19. *Id.*

20. *Id.*

21. *Campbell v. Acuff-Rose Music, Inc.*, 510 U.S. 569, 579 (1994).

22. *See Bill Graham Archives v. Dorling Kindersley Ltd.*, 448 F.3d 605, 607, 615 (2d Cir. 2006).

purpose: rather than marketing the Grateful Dead, like the original images had, the book used the images alongside explanatory text and additional artwork to chronicle the Grateful Dead's history.²³

Having found the use transformative, the court proceeded to hold that it was fair use even though some of the other factors, in isolation, arguably cut against that conclusion.²⁴ For instance, the images were clearly creative, meaning that the second fair use factor—nature of the work—favored the copyright holder.²⁵ Furthermore, the coffee-table book maker included the images in their entirety, which also typically pushes against fair use based on the third fair use factor (the amount and substantiality of the work used).²⁶ But where courts find transformative use, they frequently find fair use no matter how the second and third prongs of the fair use test come out.²⁷

The same is not entirely true of the fourth fair use factor. This factor focuses on whether the use negatively affects the market (and potential markets) for the original work.²⁸ Some view the fourth factor as the most important factor within the fair use inquiry.²⁹

But while the fourth factor remains important, courts often assess it based on how they answered the transformative use inquiry.³⁰ In other words, when a court deems a use transformative, it frequently finds that the use has no relevant impact on the original work's market. This is so because the transformative use goes beyond the original work's purpose and is thus geared toward a different market than the original work. Hence, for instance, because the Supreme Court considered 2 Live Crew's rap parody of Roy Orbison's "Oh, Pretty Woman" transformative, it concluded that the parody did not harm Orbison's market for "Oh, Pretty Woman"; the parody had a different purpose

23. *Id.* at 608-09, 615.

24. *Id.* at 615.

25. *Id.* at 612-13.

26. *Id.* at 613.

27. See Barton Beebe, *An Empirical Study of U.S. Copyright Fair Use Opinions, 1978-2005*, 156 U. PA. L. REV. 549, 605-06 (2008) ("[I]n those opinions in which transformativeness did play a role, it exerted nearly dispositive force not simply on the outcome of factor one but on the overall outcome of the fair use test.").

28. 17 U.S.C. § 107(4) (2015).

29. See, e.g., *Harper & Row, Publishers, Inc. v. Nation Enters.*, 471 U.S. 539, 566 (1985) ("This last factor is undoubtedly the single most important element of fair use.").

30. See, e.g., Beebe, *supra* note 27, at 621 ("[T]he actual doctrine of the fourth factor consists in practice of a few propositions of law that judges should keep in mind as they synthesize the various factual findings that they have made under the previous factors.").

than the original song and thus targeted a market to which Orbison's heirs were not entitled.³¹

In sum, if a court finds a use to be transformative, that finding often dictates the rest of the fair use analysis. Factors two and three are not strong enough to overcome it, and factor four, while still important, typically favors fair use because a transformative use purportedly does not negatively affect the original work's market. Conversely, if a court finds a use to be nontransformative, the fair use inquiry often moves in the opposite direction.³² Factors two and three obtain renewed status,³³ while factor four is more likely to push in favor of market harm since the nontransformative use is more likely to act as a substitute for the original work.³⁴ As such, winning the transformative use battle is of vital importance.

II. Oracle's Arguments

For this reason, Oracle and some of its amici spend a great deal of time arguing that Google's use of Oracle's Java software application programming interfaces (APIs) in Android is nontransformative. Their primary argument on this front is that Google failed to use Oracle's APIs for a new purpose.³⁵ Instead, Google uses the APIs to perform the same functions that they perform in Oracle's Java platform: to identify computing functions that other parts of the platform carry out.³⁶ In the language of the Supreme Court, Google's use merely "supersede[s] the objects[] of the original work,"³⁷ rather than adding "something new, with a further purpose or different character."³⁸

Consequently, according to Oracle and its supporters, this nontransformative use has significant negative effects, both actual and

31. *Campbell v. Acuff-Rose Music, Inc.*, 510 U.S. 569, 583, 592 (1994). The one possible exception, which the Court remanded for consideration, was with respect to non-parody rap derivatives of the song. *Id.* at 593-94.

32. *See, e.g., Sag*, *supra* note 7, at 76-77 ("[A] plaintiff can expect to win a clear majority of cases where there is no indication of transformative use, but otherwise expect to lose all but 38% of the time").

33. *See Campbell*, 510 U.S. at 579 ("[T]ransformative use is not absolutely necessary for a finding of fair use, . . . [But] the more transformative the new work, the less will be the significance of other factors . . . that may weigh against a finding of fair use."). In other words, the other fair use factors weigh more heavily when a work is not transformative. *Id.*

34. *Id.* at 584-85, 592.

35. *See, e.g., Opening Brief & Addendum for Oracle America, Inc.* at 29-37, *Oracle Am., Inc. v. Google, Inc.*, Nos. 17-1118, -1202 (Fed. Cir. Feb. 10, 2017), 2017 WL 679347.

36. *Id.* at 29-30.

37. *Campbell*, 510 U.S. at 576 (quoting *Folsom v. Marsh*, 9 F. Cas. 342, 348 (C.C.D. Mass. 1841)).

38. *Id.* at 579.

potential, on the market for Oracle's APIs. Because Google uses the APIs in its competitive Android platform but gives away Android for free, Oracle claims it is unable to find willing buyers for Java.³⁹ And Google has or is in the process of saturating a variety of computing markets with Android, including phones, tablets, desktops, wearables, cars, and others, thereby depriving Oracle of additional licensing revenues.⁴⁰ Hence, according to Oracle, Google's failure to put the APIs to a transformative use has destroyed Oracle's current and future ability to market its copyrighted APIs.⁴¹ As a result, Oracle and its supporters argue, both factors one and four of the fair use inquiry—by far the most important within that inquiry—favor Oracle.

III. Negatively Transforming Transformative Use

But Oracle's argument has several significant problems. Most importantly, it implies that fair use should have little if any applicability to software reuses. After all, software by definition is functional; software code is written to carry out specific, preassigned computing functions, so reuses of software will typically implicate the very same functions. Hence, software reuses are constrained by the functional nature of software itself. One can certainly imagine theoretical reuses of software for nonfunctional purposes (e.g., software source code as wallpaper). But those uses are rare and, even when they occur, have limited social value. Nevertheless, the implication of Oracle and its amici's arguments is that only such nonfunctional reuses of software should be eligible for a fair use defense.

In fact, carried to its extreme, this argument threatens even statutorily delineated fair use examples. As mentioned, the statute lists uses for "teaching," "scholarship," and "research" purposes as likely candidates for a fair use defense.⁴² Yet reuses of software in educational and research settings would still typically implicate a software program's functionality (i.e., the software, when used, would perform the same functions for which it was programmed). But under Oracle and its supporters' arguments, such functional uses would not qualify as transformative uses. And if not transformative uses, the uses would face extreme difficulty winning a fair use defense in modern courts for the reasons discussed above.

The far-reaching implications of Oracle's arguments can also be seen when considering how those arguments influence analysis of fair use's fourth factor. There, Oracle and its amici argue that because Google's use of the Java APIs is nontransformative, that use undermines Oracle's ability to monetize its

39. Opening Brief & Addendum for Oracle America, Inc., *supra* note 35, at 47-52.

40. *Id.* at 56-60.

41. *Id.* at 47-55.

42. 17 U.S.C. § 107 (2017).

software in every computing market.⁴³ In other words, because the functional purposes of software are identical in all computing environments, Oracle claims market harm in each and every one of those environments.

If accepted, these arguments would mean software fair use is effectively dead. But fair use plays an important role in fostering innovation and creativity and should continue to do so in the software world. The next section briefly explores ways in which courts might adapt the transformative use inquiry in the software context to ensure that fair use continues to play such a role.

IV. Reconceiving Transformative Use in Software

As mentioned, transformative use has gained momentum over the years as one of the most important factors in determining whether a particular use of copyrighted material is fair. But it is important to remember that the transformative use inquiry need not dictate the entire fair use analysis—indeed, courts have found fair use even in cases where the use was deemed nontransformative.⁴⁴ Hence, one way to solidify fair use’s position in the software reuse context is to deemphasize the importance of the transformative use prong in light of software’s unique characteristics.

For instance, factor two, the nature of the work, should carry greater weight in software fair use cases. As the Supreme Court has stated, “[t]his factor calls for recognition that some works are closer to the core of intended copyright protection than others,”⁴⁵ with the implication being that works farther from that core enjoy thinner copyright protection and are subject to a broader scope of fair use.⁴⁶ Software’s functional nature situates software farther from the core of intended copyright protection than other types of works.⁴⁷ Section 102(b) of the Copyright Act reflects this principle by making clear that despite software being eligible for copyright under Section 102(a), in no case should that protection extend to “any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied” in the software.⁴⁸ In

43. Opening Brief & Addendum for Oracle America, Inc., *supra* note 35, at 47-52, 56-60.

44. Beebe, *supra* note 27, at 605.

45. *Campbell v. Acuff-Rose Music, Inc.*, 510 U.S. 569, 586 (1994).

46. *Id.*; *Apple Comput., Inc. v. Microsoft Corp.*, 35 F.3d 1435, 1446 (9th Cir. 1994) (finding that the functional nature of Apple’s software meant that the Apple operating system was “entitled to only limited protection”).

47. See Pamela Samuelson, *Why Copyright Law Excludes Systems and Processes from the Scope of its Protection*, 85 TEX. L. REV. 1921, 1923 (2007) (“Congress intended for § 102(b) to codify the principal holdings of [*Baker v. Selden*, 101 U.S. 99 (1880)] and its progeny to limit the scope of copyright protection in functional writings, such as [computer] programs.”).

48. 17 U.S.C. § 102(b) (2015); Samuelson, *supra* note 47, at 1923.

other words, software's functional nature means that Section 102(b)'s limitations are highly relevant in restricting copyright protection in software.⁴⁹ Hence, the more limited nature of software copyright means that the scope of fair use should be broader under factor two when dealing with software reuses.⁵⁰

Courts should also more carefully consider factor three, the amount and substantiality of the work taken, in light of software's functional realities. In general, it can be difficult for courts to properly determine what constitutes the software work from which the third party borrowed (and thus how much borrowing actually occurred).⁵¹ For instance, any distinguishable part of a larger software program might be viewed as a separately copyrighted work, meaning that borrowing even just that part weighs against fair use because the party copied the copyrighted work in its entirety. In fact, in its earlier decision in *Oracle v. Google*, the Federal Circuit appeared to view the APIs in this way, i.e., as separately copyrightable works distinct from the software that implements the computing functions that the APIs specify.⁵² Viewing the APIs in that light, it becomes difficult to argue Google's use of them in Android is fair, since it borrowed the entirety of thirty-seven separate works.

Yet the APIs' functional, interdependent nature means that they have no practical reality absent the software that implements their specified functions. Unlike other types of copyrightable works, they have a solely functional purpose that is only realized once paired with other, implementing software.⁵³ Hence, courts applying this third fair use factor in software contexts should carefully assess the functional relationships of the different software components in determining the appropriate baseline against which to assess how much of the original software work was copied. In the *Oracle v. Google* case, the more appropriate unit of analysis is the larger Java platform, rather than the individual APIs themselves.

Choosing the appropriate unit of analysis under factor three also has ramifications for how courts should assess the fourth fair use factor in the software reuse context. For instance, parties often claim market harm based on a third party's use of some portion of their larger software program, but in many cases that market harm claim is dubious because the portion used has no real market of its own. This is precisely what happened in *Oracle v. Google*. Oracle claims Google's use of the APIs destroyed the market for them.⁵⁴ But that claim

49. Samuelson, *supra* note 47, at 1923.

50. Clark D. Asay, *Software's Copyright Anticommons*, 66 EMORY L.J. 265, 317 (2017).

51. See generally Margot E. Kaminski & Guy A. Rub, *Copyright's Framing Problem*, 64 UCLA L. REV. (forthcoming 2017) (discussing the general difficulty courts often have in determining what constitutes the copyrighted work).

52. See *Oracle Am., Inc. v. Google Inc.*, 750 F.3d 1339, 1367-68 (Fed. Cir. 2014).

53. Asay, *supra* note 50, at 319-22.

54. Opening Brief & Addendum for Oracle America, Inc., *supra* note 35, at 47-52.

is suspect precisely because the APIs have no real market of their own.⁵⁵ Instead, their functional, interdependent relationships to other software mean that the APIs' only real market is as part of a larger software platform such as Java or Android.⁵⁶ And while Android certainly competes with the Java platform, according to leading Ninth Circuit software fair use precedent, that type of competition is arguably permissible (and even desirable) under copyright law.⁵⁷

Software's functional nature also means that courts should carry out the transformative use inquiry itself differently in the software context than in others. Oracle and its amici, for instance, rely heavily on mostly non-software fair use cases to support their claim that reusing a work for the same purpose is not transformative and therefore not a fair use.⁵⁸ But analogizing between software and nonsoftware cases can be dangerous because software's functional characteristics make it unlike other copyrightable materials in key respects—the most important of which is that any given software component by definition has a singular computing purpose.

55. One may argue that Google's use of the APIs suggests there is, in fact, a market for them. But though courts have sometimes engaged in this type of circular reasoning, other courts have refuted it. *See, e.g., Ringgold v. Black Entm't Television*, 126 F.3d 70, 81 (2d Cir. 1997) ("Since the issue is whether the copying should be compensable, the failure to receive licensing revenue cannot be determinative in the plaintiff's favor.").

56. In fact, though Oracle's predecessor, Sun, and Google discussed a license to use the Java APIs before negotiations ultimately failed, those discussions were with respect to the entire Java platform, rather than the APIs alone. *See Order Denying Rule 50 Motions at 18, Oracle Am., Inc. v. Google Inc.*, No. C 10-03561 WHA, 2016 WL 3181206, at *11 (N.D. Cal. June 8, 2016). This reality makes sense since the APIs have no real use in isolation. The same reality also explains why Oracle has no licensing program with respect to just the APIs. Instead, they license Java-based software platforms that incorporate the APIs. *See Opening Brief & Addendum for Oracle America, Inc.*, *supra* note 35, at 48-52, where Oracle makes clear that the purported market harm is with respect to its Java-based platforms, rather than the APIs in isolation.

57. *See Sony Comput. Entm't, Inc. v. Connectix Corp.*, 203 F.3d 596, 607-08 (9th Cir. 2000) (finding fair use in spite of the defendant using the plaintiff's software to create its own, functionally-equivalent software that competed with Sony's PlayStation gaming console); *Sega Enters. Ltd., v. Accolade, Inc.*, 977 F.2d 1510, 1525 (9th Cir. 1992) (finding fair use even though the defendant used Sega's software to enable its own games to function properly on Sega's Genesis gaming console, thereby potentially competing with Sega's own games).

58. *Opening Brief & Addendum for Oracle America, Inc.*, *supra* note 35, at 29-37 (citing *Campbell v. Acuff-Rose Music, Inc.*, 510 U.S. 569, 579, 591 (1994); *Stewart v. Abend*, 495 U.S. 207, 237-38 (1990); *Gaylord v. United States*, 595 F.3d 1364, 1373-74 (Fed. Cir. 2010); *Leadsinger, Inc. v. BMG Music Publ'g*, 512 F.3d 522, 530 (9th Cir. 2008); *Perfect 10, Inc. v. Amazon.com, Inc.*, 508 F.3d 1146, 1168 (9th Cir. 2007); *Kelly v. Arriba Soft Corp.*, 336 F.3d 811, 817-19 (9th Cir. 2003); *Elvis Presley Enters., Inc. v. Passport Video*, 349 F.3d 622, 628 (9th Cir. 2003); *Worldwide Church of God v. Phila. Church of God, Inc.*, 227 F.3d 1110, 1118 (9th Cir. 2000); *Micro Star v. Formgen Inc.*, 154 F.3d 1107, 1113 & n.6 (9th Cir. 1998)).

In fact, leading Ninth Circuit case law has found that reusing software for the same basic computing functions can be transformative and, more generally, fair use.⁵⁹ In reaching this holding, the *Sony* court reasoned that, despite the “similarity of uses and functions between” the plaintiff’s and defendant’s software products, the defendant’s software was “modestly transformative” because it was a “wholly new product” consisting of significant amounts of code the defendant had written.⁶⁰ Furthermore, the defendant’s software product enabled those similar computing uses and functions in a new computing environment.⁶¹ Google’s reuse of the Java APIs in Android is similar. While Google copied thirty-seven of 166 possible Java APIs into Android, it wrote its own implementing code, virtual machine, and other APIs for Android.⁶² This “wholly new product”⁶³ also employed the APIs in an innovative smartphone platform, a venture in which Oracle and the APIs’ earlier owner, Sun Microsystems, had failed even before Google used the APIs in Android.⁶⁴

In sum, software’s functional nature should play a considerable role in how courts assess each factor within the fair use inquiry, including the transformative use component. Of course, software’s functional realities have long challenged courts to reconcile copyright law doctrines developed in other creative contexts with software’s functional realities.⁶⁵ But Oracle’s transformative use arguments, rather than attempting to reconcile fair use with software’s hybrid nature, instead seek to bring about an effective divorce between the two.

Yet the fair use inquiry, with its flexibility, provides courts with an important opportunity to achieve reconciliation. The fair use doctrine, after all, is meant to be a flexible tool that courts can use to do copyright justice.⁶⁶ In the software context, copyright justice would, at a minimum, recognize that (1) software’s functional nature means that reusing software inevitably involves performing the same functions for which the software was designed, and (2) this reality, on its own, is insufficient to foreclose a transformative use finding and, more generally, a fair use finding. Instead, software’s functional

59. *Sony*, 203 F.3d at 609.

60. *Id.* at 606.

61. *Id.*

62. Oracle Am. Inc. v. Google Inc., 872 F. Supp. 2d 974, 978-79 (N.D. Cal. 2012), *aff’d in part, rev’d in part*, 750 F.3d 1339 (Fed. Cir. 2014).

63. *Sony*, 203 F.3d at 606.

64. Larry Dignan, *Google: Oracle, Sun Blew It on a Java Smartphone*, CNET (Apr. 18, 2012, 5:46 AM PDT), <http://www.cnet.com/news/google-oracle-sun-blew-it-on-a-java-smartphone>.

65. See generally Pamela Samuelson et al., *A Manifesto Concerning the Legal Protection of Computer Programs*, 94 COLUM. L. REV. 2308 (1994) (reviewing software’s unique characteristics and concluding on the basis of them that software should have a sui generis legal regime).

66. See, e.g., Glynn S. Lunney, Jr., *Fair Use and Market Failure: Sony Revisited*, 82 B.U. L. REV. 975, 977 (2002).

nature means that transformative software reuses can and in most cases will involve reperforming the very same computing functions for which the software was originally designed. In what computing context, in conjunction with what other content and technology, should then become the more pressing questions.

Conclusion

Software reuse is an important means by which to spur robust software innovation. Copyright law's fair use defense is one important means of enabling such reuse. Yet parties such as Oracle are pressing fair use arguments that, if accepted, would largely eliminate fair use as a viable defense in the software context. This Essay urges courts to reject such arguments and instead apply fair use as intended—in a flexible way that promotes, rather than inhibits, innovation and creativity.