MEMORANDUM OF LAW[1]

To: Mark Smith, Acme, Inc.
From: Atlas Legal Research, LP
Date: June 16, 2005
Re: "Derivative work" of a software product

You have asked us to prepare a memorandum of law on the following facts and issues:

Facts:

Acme is designing a software program that will "interoperate" with and integrate into another software package not designed and/or owned by Acme. You have asked Atlas to assume that a third party holds the copyrights on the non-Acme software. However, Acme has been able to freely access the non-Acme software.

Issue:

Based on U.S. copyright statutes and relevant case law, what are the characteristics of software that would likely be deemed to be a "derivative work" of another for purposes of acquiring federal copyright protection?

Short Answer:

Merely ensuring interoperability with another program will not protect a computer program from being dubbed a derivative work of another. In order to constitute a derivative work, a program should be "based upon" another – meaning that there should be copying of either: (1) the source code – in qualitative or quantitative terms – or (2) the structure or arrangement of another program. This copying assumes greater significance when the two programs perform similar functions. Each case is fact-specific and for determining whether a program is a derivative work of another, their similarities should be assessed in detail and if there is "substantial" similarity, it would be considered a derivative work.

Analysis:

Federal statutes provide a logical starting point in understanding the boundaries of what constitutes a derivative work. A derivative work is statutorily defined as "a work based upon one or more preexisting works, such as a translation, . . . abridgment, condensation, or any other form in which a work may be recast, transformed, or adapted," including "[a] work consisting of editorial revisions, annotations, elaborations, or other

[1] This memo has been redacted and partially reformatted to protect client confidentiality.

modifications which, as a whole, represent an original work of authorship."[2] The owner of a copyright enjoys the exclusive right to make derivative works with respect to the copyrighted work.[3] Therefore, a person who makes unauthorized derivative works of a copyrighted work would be liable for infringement.[4] If, however, derivative works are made with authorization, the maker would enjoy copyrights limited to such derivative work, subject to fulfilling other requirements such as originality and fixation.[5]

Quantity vs. Quality

Because the statutory definition of a derivative work is general in its language, it is necessary to analyze how different courts have interpreted the definition in the context of computer software. Significant among these decisions is the Fifth Circuit's ruling in *Vault Corp. v. Quaid Software Ltd.*[6] Citing the Ninth Circuit's analysis in *Litchfield v. Spielberg,*[7] the Fifth Circuit held that to constitute a derivative work, "the infringing work must incorporate in some form a portion of the copyrighted work" and that "the infringing work must be substantially similar to the copyrighted work." Under the facts of *Vault Corp.*, the Court held that there was no derivative work because the two programs shared a quantitatively minor amount of source code (about 30 characters of source code were common in two programs running into 50 and 80 pages of source code). Further, the two programs served opposing functions. Thus, the Court considered both the literal and functional aspects of the software while testing whether one program was a derivative work of another.[8]

The same approach was followed in the case of *Central Point Software, Inc. v. Nugent,*[9] where a federal district court in Texas held that "[t]o constitute a derivative work, the work need only contain a substantial amount of material from the copyrighted work." It described "derivative" as a program that served the same basic function as and incorporated over 50% of the original code of another copyrighted program.

While determining whether a program is a derivative work of another, the Third Circuit noted that "[m]uch more significant, however, than the quantity of copy is the *quality* of the material purloined."[10] If the material copied is quantitatively small but

---

[2] 17 U.S.C.A. § 101.
[3] 17 U.S.C.A. § 106.
[4] 17 U.S.C.A. § 501.
[5] 17 U.S.C.A. §§ 102 & 103. *See Dynamic Solutions, Inc. v. Planning & Control, Inc.*, 646 F.Supp. 1329, 1340 (S.D.N.Y. 1986) (holding that copyright may be denied for a derivative work if it uses pre-existing material unlawfully).
[6] 847 F.2d 255, 267-268 (5th Cir.1988).
[7] 736 F.2d 1352, 1357 (9th Cir.1984), *cert. denied,* 470 U.S. 1052.
[8] *See also Qad. Inc. v. ALN Associates, Inc.*, 974 F.2d 834 (7th Cir.1992) (discussing the issue of similarity in the functions of two programs)
[9] 903 F.Supp. 1057, 1060 (E.D.Tex.1995).
[10] *Dun & Bradstreet Software Services, Inc. v. Grace Consulting, Inc*., 307 F.3d 197, 208 (3rd Cir.2002) (emphasis in original).

qualitatively substantial, a software program would be held to be a derivative work of another.[11]

The Fifth Circuit and Third Circuit decisions seem to agree that the functionality of the software programs at issue plays an important role in determining derivative work status. The Fifth Circuit, however, seems to place added emphasis on the relative quantities of identical source code. The safest approach, therefore, in avoiding derivative work status appears to be minimizing both the amount of copied source code and the overlap in functionality between the two programs.

The "Based Upon" Approach

Various courts have held that in order to constitute a derivative work, a program should be "based upon" another copyrighted program - - i.e., there should be substantial similarity between the two works.[12] However, as pointed out in *SAS Institute, Inc. v. S & H Computer Systems, Inc.*,[13] "[t]he question of the substantiality of the similarity is a question of fact" and in each of these cases, the question of similarity is addressed by a detailed factual comparison of the programs at issue.

In order to assess whether a program is "based upon" another, courts rely on the same evidentiary record and adopt the same analytic approach as when assessing an issue of infringement by copying.[14] The Ninth Circuit held in *Micro Star v. Formgen Inc.*[15] that "[a] work will be considered a derivative work only if it would be considered an infringing work if the material which it has derived from a preexisting work had been taken without the consent of a copyright proprietor of such preexisting work."

In *Q-Co Industries, Inc. v. Hoffman,*[16] a New York federal district court noted that while assessing similarity, the structure and arrangement of the computer program are also considered. However, similarity in an idea - rather than its expression – does not render one software program a derivative of another.[17] Further, in *Hoffman,*[18] the court held that "[t]he extent of similarity between the two programs is more significant" than the aspects that are dissimilar. Thus, even if there are significant original portions in the

---

[11] *SAS Institute, Inc. v. S & H Computer Systems, Inc.*, 605 F.Supp. 816, 829-831 (M.D.Tenn.1985).
[12] *S.O.S., Inc. v. Payday, Inc.*, 886 F.2d 1081 (9th Cir.1989). *Integral Systems, Inc. v. Peoplesoft, Inc.*, No. C-90-2598-DLJ., 1991 WL 498874, at *12 (N.D.Cal. Jul. 19, 1991). *Micro Consulting, Inc. v. Zubeldia* , 813 F.Supp. 1514, 1531 (W.D.Okla.1990) aff'd *Micro Consulting, Inc. v. Zubeldia*, 959 F.2d 245 (10th Cir.1992).
[13] 605 F.Supp. 816, 829-831 (M.D.Tenn.1985).
[14] *See Integral Systems, Inc. v. Peoplesoft, Inc.*, No. C-90-2598-DLJ., 1991 WL 498874, at *12 (N.D.Cal. Jul. 19, 1991).
[15] 154 F.3d 1107, 1112 (9th Cir.1998) (internal quotation marks omitted).
[16] 625 F.Supp. 608, 615-616 (S.D.N.Y.1985).
[17] *Id.*
[18] 625 F.Supp. 608, 615-616 (S.D.N.Y.1985).

program, if it is substantially and pervasively "based upon" another, it would amount to a derivative work.[19]

The cases incorporating the "based upon" approach add another dimension to determining derivative work status. Under the qualitative/quantitative analysis, both functionality and amount of overlapping source code were important. Under the "based upon" approach, even the structure and arrangement – i.e., the organization – of a computer program can assist in determining whether that program is derivative of another. This means that a programmer – when considering all the analytical paradigms – must be cognizant of the amount of identical source code, the function of the two programs, and organization of the two sets of source code.

Interoperability

The case law does not directly address whether a computer program becomes a derivative work when it ensures interoperability with another program. However, in *Dun & Bradstreet Software Services, Inc. v. Grace Consulting, Inc.*,[20] the Third Circuit rejected the doctrine of interoperability as a justification for copying though it could be a factor for determining whether certain aspects of the pre-existing work are copyrightable. We cannot determine the impact of this holding to the present inquiry because we do not know what the pre-existing program contains and what Acme might incorporate from that program. Nonetheless, substantial similarity could render one program a derivative work of another, even if the similarity is for the purpose of achieving interoperability.

Conclusion:

Clearly, the courts have taken varying approaches to determining whether a computer program is a derivative work of another. If Acme's goal is to reconcile all the approaches in an effort to minimize the possibility of its program being deemed a derivative work of another, then Acme should be mindful of the following factors:

- the physical amount of source code that is identical to the original program;
- the functions served by the two programs, with the goal being to create as much divergence as possible;
- the similarity in organization between the two sets of code; and
- the degree to which the copied source code is copyrightable where the two programs interoperate.

[19] *SAS Institute, Inc. v. S & H Computer Systems, Inc.*, 605 F.Supp. 816, 829-831 (M.D.Tenn.1985).
[20] 307 F.3d 197, 208 (3rd Cir.2002).

How each of these factors will affect a specific analysis will depend on the underlying facts. For the present inquiry, we were given very general facts, and our assumption has been that Acme wants a general roadmap on the doctrine of derivative works. We will, of course, be happy to undertake a more detailed analysis when provided with specific facts as to a particular software program.

We hope this helps, Mark. We also thank you for reposing your confidence in us. Though this is our first project for Acme, we hope that it will be the first of many. As you know, Atlas Legal Research, LP is not a law firm, and our findings cannot constitute a legal opinion. Please have this analysis reviewed by appropriate in-house and/or outside legal counsel before making any final conclusions.

Should you have any further questions or require any follow-up, please call us at 866-52-ATLAS, or email Rocky Dhir at rocky@atlaslegal.com. We look forward to serving you again soon.