Shrink-Wrap Licenses and Reverse Engineering

2003-04-16

Background

ProCD v. Zeidenberg is a leading case on the enforceability of shrink-wrap licenses as typically provided with packaged software. See our July 1996 Intellectual Property Bulletin for more on the *ProCD v. Zeidenberg* case. In that case, ProCD sold a compact disc with a compilation of non-copyrightable "white pages" data. Before a user could access the data, the user was required to accept the terms of a license agreement that prohibited the licensee from using the product for commercial use. Zeidenberg posted the data on a website on which he was also selling advertising. The court enforced the license restriction against commercial use, even though the licensed data was not copyrightable.

In *Bowers v. Baystate Technologies, Inc.*, a recent decision by the United States Court of Appeals for the Federal Circuit (CAFC), the court applied Massachusetts law and held that a clause prohibiting reverse engineering in a shrink-wrap license was enforceable against a licensee. The CAFC reached this result, even though it had decided in an earlier case that reverse engineering was a "fair use" of copyrightable material under the Copyright Act. The CAFC stated that reverse engineering supported the Constitutional objective of "encourag[ing] others to build freely upon the

ideas and information conveyed by a work."

The *Bowers* decision is important because software developers who reverse engineer another company's products to develop their own software or to make their software interoperable with other software may be found liable for violating the terms of that other company's shrink-wrap license. Depending on how broadly the *Bowers* case is followed, a shrink-wrap license could, in effect, take on characteristics of a patent and/or a covenant not to compete.

In contrast, the European Union in its 1991 Software Directive recognizes a much broader right to reverse engineer. The Software Directive provides that the licensee shall be entitled to reverse engineer a software program, if the information necessary to achieve the interoperability of that software program with other programs has not previously been made readily available to the licensee. This right is provided by statute, and cannot be waived contractually.

The Bowers Case

In *Bowers,* the CAFC affirmed the trial court's ruling that Baystate violated the terms of Bowers' shrink-wrap license by reverse engineering features implemented in a software product referred to as a computer aided design (CAD) template that Bowers commercialized under the name "Designer's Toolkit." Baystate obtained several copies of the Designer's Toolkit, and then introduced a competing product that had features that were similar to those of Bowers' Designer's Toolkit.

The question before the court was the extent to which a provider of copyrighted material could use a shrink-wrap license to override the ability

of a party to reverse engineer software under the fair use doctrine of copyright law.

In a split decision (2-1), the majority stated that both parties freely entered into the terms of the shrink-wrap license, and concluded that private parties are free to contractually forego reverse engineering privileges set forth by the Copyright Act, thus following the reasoning of the Seventh Circuit in *ProCD v. Zeidenberg.*

The CAFC used a broad dictionary definition of reverse engineering: "to study or analyze (a device, as a microchip for computers) in order to learn details of design, construction, and operation, perhaps to produce a copy or an improved version." While the primary purpose of clauses banning reverse engineering is often to prevent one from viewing source code considered to be a trade secret, the definition that was used in this case is broader.

In its analysis, the CAFC held that the trial court incorrectly required Bowers to show that Baystate infringed Bowers' copyright in order to find that there was reverse engineering. In the trial court, Bowers presented evidence, and the jury agreed, that Baystate had infringed its copyright.

In the CAFC's analysis, however, it looked at the similarities in the programs in an analysis that was similar to a copyright analysis to determine that there must have been reverse engineering.

Future cases will determine the scope of such restrictions against reverse engineering. Could such a restriction be used to prohibit a party from licensing software and analyzing its functions simply from a user perspective, without looking at source code? Or could it prohibit a reviewer from

analyzing and reviewing the features of the software? These scenarios might go too far, but the court's approach would not foreclose the possibility. A court could require more activity by a licensee to constitute reverse engineering, such as a detailed analysis prepared for competitive purposes.

If just looking and analyzing were included in reverse engineering, a clause prohibiting reverse engineering would essentially allow a licensor to protect information that most would consider publicly available, as though such information were a trade secret. Moreover, a licensor could then effectively limit the ability of a third party from competing with a product with similar functions, even though the protection of publicly viewable methods would typically be obtained through patent protection after an examination process. In this case, Bowers did have a patent, but the CAFC ruled that the patent was not infringed by Baystate.

Conclusion

Reverse engineering fosters competition by allowing other parties to identify, add, and improve on features that are in other parties' software programs. There are still limitations to what a reverse engineer can do: it cannot infringe patent rights, which can cover the functions that are performed, and it cannot copy software code, which would be a violation of copyright rights.

The *Bowers* decision makes clear that "no reverse engineering" clauses that are typical in shrink-wrap licenses are enforceable, even if the use would be a fair use under the copyright law. The court did not define the limits of what constitutes reverse engineering. Depending on how such clauses are worded and enforced, they could limit the ability of third parties to use similar processes and functions to develop competing programs, or to

maintain interoperability between programs, thus going beyond copyright law and approaching the broader types of protection secured through patent law.

Corey Scott

corey.scott@haledorr.com

Michael A. Diener

michael.diener@haledorr.com