Westlaw Journal
INTELLECTUAL PROPERTY
Litigation News and Analysis • Legislation • Regulation • Expert Commentary
VOLUME 24, ISSUE 2 / MAY 10, 2017

EXPERT ANALYSIS

Using Open Source Code for Development of 'Proprietary' Software

By Nancy A. Del Pizzo, Esq.
Rivkin Radler LLP

"Open source" commonly refers to software "whose code is made freely available to all users" on the theory that a "crowd-sourced product, where the public is permitted to modify the code, "will be superior to a privately developed" product.[1]

Software developers may use open source code to help them more efficiently develop new software products, which can reduce the cost of their services.

But what sounds like a probable benefit to companies that rely on software development could result in legal liability. This is because companies that employ software developers or contract with them to develop proprietary programs may not be aware that open source code, while free to use, may render their "proprietary" products unprotectable.

That's because code provided via "open source" is typically made available under a license that comes with restrictions — even though it allows for free use and distribution.

Those restrictions can be quite rigid. For example, they may include a requirement that any new program using the open source code must become open source as well.

In fact, there are more than 100 different licenses associated with open source code.[2] Some require that any products made using the code also be provided as open source.

Others mandate that any new work developed using the code include a copyright notice acknowledging the author of the open source code used.

Still others require the user to bear the risk of any errors in the code — or liability for infringement of a third party's right to it.

Moreover, some licenses contradict others. This means that if more than one type of open source code is used in a program, it may be impossible to comply with the terms of one of the licenses.

Thus, it is incumbent on anyone who plans to use open source code to read and understand the license or licenses under which the open source code is provided.

It is also critical to require employees and contracted software developers to reveal the sources of the code, as well as all applicable licenses.

LIABILITY FOR NONCOMPLIANCE WITH LICENSES

It has long been held that if a party uses open source code and does not comply with the license for its use, the causes of action available to the owner are not restricted to breach of contract.[3]

The fact that open source code is made available at no cost does not render restrictions in a license meaningless.

In fact, the copyright holder of the open source code can grant a right to make certain modifications to the code while retaining the right to restrict other modifications.

There are more than 100 different licenses associated with open source code.

The open source licensor may also require the user to include a copyright notice if the user distributes code copies.

Potential harm includes more than just monetary loss. It may also include a market share the program creator may obtain by providing components of software free to the public or an increase in the programmer's international reputation.

It is possible for the alleged harm to be based exclusively on a theory of copyright infringement.

Thus, licenses provided to users of open source code are not identical. They should be thoroughly read and understood prior to using the code.

In fact, infringement claims — along with breach-of-contract claims — may be based on the terms of the license.

In *Software Freedom Conservancy Inc. v. Best Buy Co.*, No. 09-cv-10155, 2010 WL 2985320 (S.D.N.Y. July 27, 2010), the plaintiff provided open source code for free. Its user license, a GNU General Public License, permitted users to modify, copy and redistribute its code subject to the terms of the license.

The plaintiff argued that the defendants distributed its copyrighted software without its permission — specifically by distributing copies within its high-definition television products and software for use with the HDTVs "on terms other than those contained in the license."

The plaintiff notified the defendants of the alleged infringement, but the defendants ignored the notice. That decision led the court to find the defendants' acts willful when it considered the plaintiff's motion for a default judgment.

The court awarded the full amount of statutory damages, trebling that amount for the willful infringement and awarding attorney fees. It also directed forfeiture of the infringing units.

While breach-of-contract claims premised on copying or reproducing open source code in violation of an applicable license are likely preempted by the U.S. Copyright Act, claims based on license terms that do not fall under copyright law are not similarly preempted.

For instance, in *Versata Software Inc. v. Ameriprise Financial Inc.*, No. 14-cv-12, 2014 WL 950065 (W.D. Tex. Mar. 11, 2014), the license accompanying the open source code required that any derivative works created using the open source code were to be made available for free as open source.

The court found that this component of the license was "separate and distinct from any copyright obligation" because "copyright law imposes no open source obligation" and the plaintiff had sued the defendant based on a breach of an additional obligation: namely, "an affirmative promise to make its derivative work open source."

TRADE SECRET CLAIMS AVAILABLE

In *Computer Associates International v. Quest Software Inc.*, 333 F. Supp. 2d 688 (N.D. Ill. 2004), the plaintiff secured a preliminary injunction against the defendant based on allegations of copyright infringement and trade secret misappropriation related to its computer source code.

The plaintiff company acquired Platinum Technology International Inc., the developer of a software program called enterprise database administrator, or EDBA. The program was designed to allow database administrators to automate tasks.

A few of the defendant's employees had played a role in developing EDBA when they were employees of Platinum.

Those employees allegedly referenced source code in EDBA when they developed a similar program for the defendant.

Citing *Unix Systems Laboratories Inc. v. Berkeley Software Design Inc.*, 27 U.S.P.Q.2d 1721 (D.N.J. 1993), the court found that the "entire EDBA source code may qualify for trade secret protection" even though some of its components — namely, "parsers, analysis engines and job schedulers" — are commonly used and well-known in programming.

The basis for this conclusion was that while they were well-known generally, the evidence indicated that those components were not "widely used in database administration software."

The court indicated that even if pieces of code are publicly available, the organization of that code may be protected as a trade secret if the code has not been disclosed as a whole.

The defendants attacked the copyright infringement claim by arguing that some of the code used in EDBA was from open sources and thus was not copyrightable.

But the court found that although parts in the public domain were not copyrightable, modifications to that third-party code, and the finished product, could be.

An IP lawyer might argue that the final result would be a non-protectable derivative work.

The court explained that the plaintiff's final product was not an unprotectable derivative work for several reasons.

First, the functions of the plaintiff's software were not essentially the same as the open source code it used.

The plaintiff used "elements of preexisting code in creating EDBA," but the use and the final product "was an original creation."

The court also noted that an open source program used to create EDBA, called Bison, was subject to the terms of a GNU General Public License.

The GPL expressly restricted permissible modifications and intended that code to remain free. According to the defendants, this meant that the plaintiff violated the GPL by claiming copyright in a program containing Bison code.

The court disagreed because EDBA used Bison to create a "utility program to create the parser source code."

Thus, the plaintiff modified Bison to suit the specific task of creating a parser for use in its database administration software, the court said.

The court stated that the GPL prevented the plaintiff from claiming a copyright in its modified version of Bison.

But it added that "the output of that program" was not subject to the GPL restrictions, particularly because of an exception in the GPL that provided that when the "file is copied by Bison into a Bison output file," the user could "use that output file without restriction."

The defendant also argued that the plaintiff had committed copyright fraud or misuse because the plaintiff did not list all of the third-party software it used in its copyright applications.

However, the court said that kind of disclosure is required only if a party uses "a substantial amount of a preexisting work." In this case, the defendant's own expert found third-party work represented less than 5 percent of the plaintiff's source code.

OPEN SOURCE AND PATENT DISCLOSURES

Open source code can also create issues in a patent context.

Consider *Implicit Networks Inc. v. Hewlett-Packard Co.*, No. 10-cv-3746, 2011 WL 3954809 (N.D. Cal. Sept. 7, 2011), a patent infringement case from the U.S. District Court for the Northern District of California.

There, the parties were obligated to serve certain disclosures under the local patent rules. The defendant argued that the plaintiff did not comply with this obligation because it failed to meet the reverse engineering or equivalent standard under the local rules.

The defendant noted that since its product was based on open source software, which is provided to customers at no charge, anyone, including the plaintiff, could obtain a copy on their own.

Therefore, the defendant said, the plaintiff should be required to reverse engineer the product and specifically identify where the purported patent infringement occurred.

The court agreed with the defendant. Since the product's code was free and available, the plaintiff was required to reverse engineer the alleged infringing product, and not the open source code it was allegedly based on, the court said.

NOT NECESSARILY WITHOUT COST

Yes, there are benefits to using open source code to develop new software programs, including efficiency and cost savings.

But where a company intends to distribute, manufacture or sell the software product derived, in part, from open source code, it is imperative to recognize that "free" code does not necessarily come without cost.

Carefully reviewing any and all licenses under which the code is provided — and demanding that contractors identify the open source code they used and provide copies of those licenses — will enable a proper assessment of the risk that may come with use of the code for an intended proprietary product.

NOTES

[1] *Versata Software Inc. v. Ameriprise Fin. Inc.*, No. 14-cv-12, 2014 WL 950065 (W.D. Tex. Mar. 11, 2014).

[2] *Open Source Licenses by Category,* Open Source Initiative, http://bit.ly/209QBEh (last visited May 8, 2017).

[3] *Jacobsen v. Katzer,* 535 F.3d 1373 (Fed. Cir. 2008) ("copyright holders who engage in open source licensing have the right to control the modification and distribution of copyright material" (citing *Gillam v. ABC Inc.,* 538 F.2d 14 (2d Cir. 1976)).

Nancy A. Del Pizzo is a partner at **Rivkin Radler LLP** in Hackensack, New Jersey. Her practice focuses on intellectual property and commercial litigation as well as privacy, data and cyberlaw. She can be reached at nancy.delpizzo@rivkin.com.