Peer-to-Peer File Sharing and Copyright Law: A Primer for Developers

by Fred von Lohmann
Senior Intellectual Property Attorney
Electronic Frontier Foundation
fred@eff.org

I. What this is, and who should read it

The future of peer-to-peer file-sharing and related technologies is entwined, for better or worse, with copyright law. If the early legal skirmishes yield any lesson for P2P developers, it is that an appreciation of the legal environment should be part of any development effort from the beginning, rather than bolted on at the end.

This piece is meant as a general explanation of the U.S. copyright law principles most relevant to P2P file-sharing technologies. It is aimed primarily at:

- Developers of core P2P file-sharing technology, such as the underlying protocols, platform tools, and specific client implementations; and
- Developers of ancillary services that depend upon or add value to P2P file-sharing networks, such as providers of search, security, metadata aggregation, and other services.

This paper is aimed not at giving you all the answers, but rather at allowing you to recognize the right questions to ask.[1]

What this is not: The following discussion focuses only on U.S. copyright law. While non-copyright principles may also be mentioned, this discussion does not attempt to examine other legal principles that might apply to P2P file-sharing, including patent, trademark, trade secret, or unfair competition. Nothing contained herein constitutes legal advice—please discuss your individual situation with your own attorney.

[1] A longer version of this paper, updated from time to time, is available at www.eff.org.

II. Copyright Basics and the Intersection with P2P File-sharing

The nature of digital file-sharing technology inevitably implicates copyright law. First, since every digital file is "fixed" for purposes of copyright law (whether on a hard drive, CD, or merely in RAM), the files being shared generally qualify as copyrighted works. Second, the transmission of a file from one person to another results in a reproduction, a distribution, and possibly a public performance (in the world of copyright law, "public performance" includes the act of transmitting a copyrighted work to the public). To a copyright lawyer, every unauthorized reproduction, distribution, and public performance requires an explanation, and thus file-sharing systems seem suspicious from the outset.

A. The end-users: "direct" infringement.

For the end-users who are sharing files, the question becomes whether the reproductions, distributions, and public performances are authorized by the copyright owner or otherwise permitted under copyright law (as "fair use" for example). If not, the end-users are what copyright lawyers call "direct infringers"—they have directly violated one or more of the copyright owner's exclusive rights.

In a widely-used public peer-to-peer file-sharing environment, it is a virtual certainty that at least some end-users are engaged in infringing activity (unless specific technical measures are taken to prevent this, like permitting only the sharing of files that have been cryptographically marked as "authorized").

B. The P2P tool maker: "contributory" and "vicarious" infringement

But what does this have to do with those who develop and distribute peer-to-peer file-sharing tools? After all, in a pure peer-to-peer file-sharing network, the developer of the file-

sharing tool has no direct involvement in the discovery, copying or transmission of the files being shared.

Copyright law, however, sometimes reaches beyond the direct infringer to those who were only indirectly involved. As in many other areas of the law (think of the "wheel man" in a stick up, or supplying a gun to someone you know is going to commit a crime), copyright law will sometimes hold one individual accountable for the actions of another. So, for example, if a swapmeet owner rents space to a vendor with the knowledge that the vendor sells counterfeit CDs, the swapmeet owner can be held liable for infringement alongside the vendor.

This indirect, or "secondary," liability can take two distinct forms: contributory and vicarious.

1. Contributory Infringement

Contributory infringement is similar to "aiding and abetting" liability: "one who, with knowledge of the infringing activity, induces, causes, or materially contributes to the infringing conduct of another, may be held liable as a contributory infringer." In order to prevail on a contributory infringement theory, a copyright owner must prove:

Direct Infringement: There has been a direct infringement by someone.

Knowledge: The accused contributory infringer knew of the underlying direct infringement. This element can be satisfied by showing either that the contributory infringer *actually* knew about the infringing activity, or that he reasonably *should have known* given all the facts and circumstances. At a minimum, however, the contributory infringer must have some specific information about infringing activity—the mere fact that the system is capable of being used for infringement, by itself, is not enough.

Material Contribution: The accused contributory infringer induced, caused, or materially contributed to the underlying direct infringement. Merely providing the "site and facilities" that make the direct infringement possible can be enough.

2. Vicarious Infringement

Vicarious infringement is derived from the same legal principle that holds an employer responsible for the actions of its employees. A person will be liable for vicarious infringement if he has the right and ability to supervise the direct infringer and also has a direct financial interest in his activities. Thus, in order to prevail on a vicarious infringement theory, a copyright owner must prove each of the following:

Direct Infringement: There has been a direct infringement by someone.

Right and Ability to Control: The accused vicarious infringer had the right and ability to control or supervise the underlying direct infringement. This element does not set a high hurdle. For example, the Napster court found that the ability to terminate user accounts or block user access to the system was enough to constitute "control."

Direct Financial Benefit: The accused vicarious infringer derived a "direct financial benefit" from the underlying direct infringement. In applying this rule, however, the courts have not insisted that the benefit be especially "direct" or "financial"—almost any benefit seems to be enough. For example, the Napster court found that "financial benefit exists where the availability of infringing material acts as a draw for customers" and the growing user base, in turn, makes the company more attractive to investors.

The nature of vicarious infringement liability creates a strong incentive to monitor the conduct of users. This stems from the fact that knowledge is not required for vicarious infringement liability—a person can be a vicarious infringer even if they are completely unaware of infringing activity.

In other words, if you exercise control over your users and derive a benefit from their activities, you remain ignorant of their conduct at your own risk.

III. Potential Defenses to Contributory and Vicarious Liability

A. No Direct Infringer: "All of My Users are Innocent"

If there is no direct infringement, there can be no indirect liability. Consequently, if a peer-to-peer developer can plausibly claim that no users in the network are sharing copyrighted works without authorization, this would be a complete defense to any contributory or vicarious infringement claims.

Unfortunately, this may be extremely difficult to demonstrate, given the decentralized nature of most P2P networks and the wide variety of uses to which they may be put. It will likely be difficult to show that *every* user is innocent. Nevertheless, in certain specialized networks that permit the sharing of only secure, authorized file types, this may be a viable defense.

B. The *Betamax* Defense: "Capable of substantial noninfringing uses"

Holding technology developers responsible for the unlawful acts of end-users obviously can impose a crushing legal burden on those who make general-purpose tools. Fortunately, the Supreme Court has defined an outer limit to copyright's indirect liability theories.

In a case involving the Sony Betamax VCR, the Supreme Court found that contributory infringement liability could not reach the manufacturer of a device that is "capable of substantial noninfringing use." In that case, the Court found that the VCR was capable of several noninfringing uses, including the time-shifting of television broadcasts by home viewers. In the Court's view, it does not matter what proportion of the uses are noninfringing, only whether the technology is "capable" of substantial noninfringing uses.

Unfortunately, the "*Betamax* defense" has been under sustained legal attack in the cases involving P2P technology. In the *Napster* case, the court found that this defense does not apply at all to vicarious liability. Accordingly, if you have control over, and derive a financial benefit from, direct infringement, the existence of "substantial noninfringing uses" for your service is irrelevant.

Moreover, the *Napster* court concluded that the *Betamax* defense may only apply until the copyright owner notifies you regarding specific infringing activity by end-users. At that point, a failure to act to prevent further infringing activity will give rise to liability, and the existence of "substantial noninfringing uses" becomes irrelevant.

The "*Betamax* defense" has also come under attack in the *Aimster* case, where a court stated that the defense was not available where the technology is primarily used for infringement. (This notwithstanding the fact that the "proportion of uses" test was explicitly rejected in the Supreme Court's *Betamax* ruling.) The scope of the "*Betamax* defense" is also at the heart of the case against Kazaa, Morpheus and Grokster, currently pending in Los Angeles.

The recent court interpretations of the "*Betamax* defense" have at least two important implications for P2P developers. First, it underscores the threat of vicarious liability—at least in the Ninth Circuit, a court will not be interested in hearing about your "substantial noninfringing uses" if you are accused of vicarious infringement. Accordingly, "control" and "direct financial benefit," as described above, should be given a wide berth.

This will likely reduce the attractiveness of business models built on an on-going "service" or "community-building" model, to the extent that these models allow the provider to control user activity (i.e., terminate or block users) and create value by attracting a large user base.

Second, with respect to contributory infringement, the recent interpretations of the *Betamax* defense suggest that, once you receive specific notices from copyright owners about infringing activities, your "substantial noninfringing uses" may no longer serve as a complete shield to contributory liability. The risk then arises that a developer may have a legal duty to "do something" about the infringing activities.

But what "something" must be done? The *Napster* decision recognizes that the ability to

respond to these notices may be limited by the technology behind the challenged service or product. In cases involving decentralized P2P networks, there may be nothing a software developer can do to stop future infringements (just as Xerox cannot control what a photocopier is used for after it is sold).

Nevertheless, copyright owners are arguing that technologists should have a duty to redesign technologies once they are put on notice regarding infringing end-users. What this might entail is difficult to predict, but may include, in some cases, modification of the architecture and capabilities of the tool, service or system.

The exact contours of the *Betamax* defense are still being developed in the courts, some of which seem to have embraced conflicting interpretations. Breaking developments on this front may have important ramifications for P2P developers and should be closely monitored.

C. DMCA Section 512 "safe harbors"

In 1998, Congress enacted a number of narrow "safe harbors" for copyright liability. These safe harbors appear in section 512 of the Copyright Act (*see* 17 U.S.C. § 512). These safe harbors apply only to "online service providers," and only to the extent that the infringement involves four functions: (1) transitory network transmissions; (2) certain kinds of caching; (3) storage of materials on behalf of users (e.g., web hosting, remote file storage); and (4) the provision of information location tools (e.g., providing links, directories, search engines).

Because Congress did not anticipate P2P when it enacted the safe harbors, many P2P products may not fit within the four enumerated functions. For example, according to an early ruling by the district court in the *Napster* case, an OSP cannot use the "transitory network transmission" safe harbor unless the traffic in question passes through its own private network. Many P2P products will, by their very nature, flunk this requirement, just as Napster did.

In addition to being limited to certain narrowly-circumscribed functions, the safe harbors are only available to entities that comply with a number of complex, interlocking statutory requirements.

In the final analysis, qualifying for any of the DMCA safe harbors requires careful attention to the legal and technical requirements and obligations that the statute imposes. Any P2P developer who intends to rely on them should seek qualified legal counsel at an early stage of the development process—an after-the-fact effort to comply is likely to fail (as it did for Napster).

IV. Lessons and Guidelines for P2P Developers

Because the relevant legal principles are in flux, these guidelines represent merely one, general analysis of the legal landscape—please consult with an attorney regarding your particular situation.

A. Make and store no copies.

This one may be obvious, but remember that if you make or distribute any copies (even if only in RAM) of copyrighted works, you may be held liable as a direct infringer. The court will not be interested in "control" or "knowledge" or "financial benefit" or "material contribution." If you made or transmitted copies, you're probably liable for infringement.

Of course, this shouldn't be a problem for most P2P developers, since the great insight of peer-to-peer architectures is that the actual resources being shared need not pass through any central server. Nevertheless, be careful where caching or similar activities are concerned.

B. Your two options: total control or total anarchy.

In the wake of recent decisions on indirect copyright liability, it appears that copyright law has foisted a binary choice on P2P developers: either build a system that allows for thorough monitoring and control over user activities, or build one that makes such monitoring and control completely impossible.

Contributory infringement requires that you have "knowledge" of, and "materially contribute" to, someone else's infringing activity. In most cases, it will be difficult to avoid "material contribution"—after all, if your

system adds any value to the user experience, a court may conclude that you have "materially contributed" to any infringing user activities.

So the chief battleground for contributory infringement will likely be the "knowledge" issue. The applicable legal standards on this question are still very much in dispute —especially as relates to the "*Betamax* defense." The *Napster* court's analysis suggests that once you receive notice that your system is being used for infringing activity (e.g., a "cease and desist" letter from a copyright owner), you have a duty to "do something" to stop it.

What might that "something" be? Well, it should be limited by the architecture of your system, but may ultimately be decided by a court. So, in order to avoid the unpleasant surprise of a court telling you to re-engineer your technology to stop your infringing users, you can either include mechanisms that enable monitoring and control of user activities (and use them to stop allegedly infringing activity when you receive complaints), or choose an architecture that will convince a judge that such monitoring and control is impossible. (Copyright owners have begun arguing that you must at least redesign future versions of your software to prevent infringement. This remarkable argument has not yet been accepted by any court.)

The *Napster* court's vicarious liability analysis also counsels for either a total control or total anarchy approach. Vicarious liability requires that you "control," and receive "benefit" from, someone else's infringing activity. The "benefit" element will be difficult to resist in many P2P cases (at least for commercial products)—so long as the software permits or enables the sharing of infringing materials, this will serve as a "draw" for users, which can be enough "benefit" to result in liability.

So the fight will likely center on the "control" element. The *Napster* court found that the right to block a user's access to the service was enough to constitute "control." The court also found that Napster had a duty to monitor the activities of its users "to the fullest extent" possible. Accordingly, in order to avoid vicarious liability, a P2P developer would be wise to either incorporate mechanisms that make it easy to monitor and block infringing users, or choose an architecture that will convince a judge that monitoring and blocking is impossible.

C. Better to sell stand-alone software products than on-going services.

Vicarious liability is perhaps the most serious risk facing P2P developers. Having the power to terminate or block users constitutes enough "control" to justify imposing vicarious liability. Add "financial benefit" in the form of a business model that depends on a large user base, and you're well on your way to joining Napster as a vicarious infringer. This is true even if you are completely unaware of what your users are up to—the pairing of "control" and "financial benefit" are enough.

Of course, most "service" business models fit this "control" and "benefit" paradigm. What this means is that, after the Napster decision, if you offer a "service," you may have to monitor your users if you want to escape liability. If you want to avoid monitoring obligations, you'll have to give up on "control."

Vendors of stand-alone software products may be in a better position to resist monitoring obligations and vicarious infringement liability. After Sony sells a VCR, it has no control over what the end-user does with it. Neither do the makers of photocopiers, optical scanners, or audio cassette recorders. Having built a device with many uses, only some of which may infringe copyrights, the typical electronics manufacturer has no way to "terminate" end-users or "block" their ability to use the device. The key here is to let go of any control you may have over your users—no remote kill switch, automatic updates feature, contractual termination rights, or other similar mechanisms.

D. Can you plausibly deny knowing what your end-users are up to?

Assuming that you have escaped vicarious infringement by eliminating "control" or "financial benefit," there is still the danger of contributory infringement. To avoid liability here, you will need to address whether you knew, or should have known, of the infringing activity of your users.

Have you built a level of "plausible deniability" into your product architecture and business model? If you promote, endorse, or facilitate the use of your product for infringing activity, you're asking for trouble. Similarly, software that sends back usage reports may lead to more knowledge than you want. Customer support channels can also create bad "knowledge" facts. Instead, talk up all the great legitimate capabilities, sell it (or give it away), and then leave the users alone. Again, your choices are total control, or total anarchy.

E. What are your substantial noninfringing uses?

If your product is intended to work solely (or best) as a mechanism for copyright piracy, you're asking for legal trouble. More importantly, you're thinking too small. Almost all peer-to-peer systems can be used for many different purposes, some of which the creators themselves fail to appreciate.

So create a platform that lends itself to many uses. Actively, sincerely, and enthusiastically promote the noninfringing uses of your product. Gather testimonials from noninfringing users. The existence of real, substantial noninfringing uses will increase the chances that you can invoke the "*Betamax* defense" if challenged in court.

F. Do not promote infringing uses.

Do not promote any infringing uses. Be particularly careful with marketing materials and screenshot illustrations—entertainment company attorneys are very good at making hay out of the fact that Beatles songs were included in sample screenshots included in marketing materials or documentation. Have an attorney review these materials closely.

G. Disaggregate functions.

Separate different functions and concentrate your efforts on a discrete area. In order to be successful, peer-to-peer networks will require products to address numerous functional needs—search, namespace management, security, dynamic file redistribution—to take a few examples. There's no reason why one entity should try to do all of these things. In fact, the creation of an open set of protocols, combined with a competitive mix of interoperable, but distinct, applications is probably a good idea from a product-engineering point of view.

This approach may also have legal advantages. If Sony had not only manufactured VCRs, but also sold all the blank video tape, distributed all the TV Guides, and sponsored clubs and swap meets for VCR users, the Betamax case might have turned out differently. Part of Napster's downfall was its combination of indexing, searching, and file sharing in a single piece of software. If each activity is handled by a different product and vendor, on the other hand, each entity may have a better legal defense to a charge of infringement.

A disaggregated model, moreover, may limit what a court can order you to do to stop infringing activity by your users. As the Napster court recognized, you can only be ordered to police your own "premises"—the smaller it is, the less you can be required to do.

Finally, certain functions may be entitled to special protections under the "safe harbor" provisions of the Digital Millennium Copyright Act ("DMCA"). Search engines, for example, enjoy special DMCA protections. Thus, the combination of a P2P file sharing application with a third party search engine might be easier to defend in court than Napster's integrated solution.

H. Don't make your money from the infringing activities of your users.

Avoid business models that rely on revenue streams that can be directly traced to infringing activities. For example, a P2P file-sharing system that includes a payment mechanism might pose problems, if the system vendor takes a percentage cut of all payments, including payments generated from sales of bootleg Divx movie files.

I. Give up the EULA

Although end-user license agreements ("EULAs") are ubiquitous in the software world, copyright owners have used them in P2P cases to establish "control" for vicarious liability purposes. On this view, EULAs represent

"contracts" between vendors and their users, and thus give software vendors legal control over end-user activities. EULAs that permit a vendor to terminate at any time for any reason may raise particular concerns, insofar as they leave the impression that a vendor has the legal right to stop users from using the software.

P2P software vendors should consider distributing their code without a EULA. Even without a EULA, a software developer retains all of the protections of copyright law to prevent unauthorized duplication and modifications.

J. No "auto-updates"

Stay away from any "auto-update" features that permit you to automatically patch, update, or otherwise modify software on the end-user's machine. Copyright owners have argued that these features establish "control" for vicarious liability purposes (on the theory that you can always "update" software to prevent its use for infringement, by retrofitting acoustic filtering, for example).

At a minimum, users should always retain the ability to decline any update. Control should rest in the end-user's hands, not the software vendor's (this as much for security reasons as legal reasons).

K. No customer support.

Any evidence that you have knowingly assisted an end-user in committing copyright infringement will be used against you. In the P2P cases so far, one source for this kind of evidence is from customer support channels, whether message board traffic or email. A user writes in, explaining that the software acted strangely when he tried to download *The Matrix*. If you answer him, copyright owners will make it seem that you directly assisted the user in infringement, potentially complicating your contributory infringement defense.

Even if you read the message but don't answer, or answer in a general FAQ, copyright owners may argue that support requests were enough to create "knowledge" of infringing activities.

So let the user community support themselves in whatever forums they like. Keep your staff out of it. (This will be easier if you are open source, of course.)

L. Be open source.

In addition to the usual litany of arguments favoring the open-source model, the open source approach may offer special advantages in the peer-to-peer realm. It may be more difficult for a copyright owner to demonstrate "control" or "financial benefit" with respect to an open source product. After all, anyone can download and compile open source code, and no one has the ability to "terminate" or "block access" or otherwise control the use of the resulting applications.

"Financial benefit" may also be a problematic concept where the developers do not directly realize any financial gains from the code (as noted above, however, the Napster court has embraced a very broad notion of "financial benefit," so this may not be enough to save you). Finally, by making the most legally dangerous elements of the P2P network open source (or relying on the open source projects of others), you can build your business out of more legally defensible ancillary services (such as search services, bandwidth enhancement, file storage, file meta-data services, etc.).

* * *

References:

A&M Records, Inc. v. Napster, Inc., 239 F.3d 1004 (9th Cir. 2001).

A&M Records, Inc. v. Napster, Inc., 2000 WL 573136 (N.D. Cal. May 12, 2000) (ruling on DMCA 512 safe harbor).

In re Aimster Copyright Litigation, 2002 WL 31006142 (N.D. Ill. Sept. 4, 2002) (granting preliminary injunction against Aimster).

MGM v. Grokster (Kazaa/Morpheus/Grokster) summary judgment briefs: http://www.eff.org/IP/P2P/MGM_v_Grokster/