MINTZ LEVIN
COHN FERRIS
GLOVSKY AND
POPEO PC

BOSTON
WASHINGTON
RESTON
NEW YORK
NEW HAVEN

www.mintz.com

One Financial Center
Boston, Massachusetts 02111
617 542 6000
617 542 2241 fax

666 Third Avenue
New York, New York 10017
212 935 3000
212 983 3115 fax

701 Pennsylvania Avenue, N.W.
Washington, D.C. 20004
202 434 7300
202 434 7400 fax

11911 Freedom Drive
Reston, Virginia 20190
703 464 4800
703 464 4895 fax

157 Church Street
New Haven, Connecticut 06510
203 777 8200
203 777 7111 fax

Please contact James G. Gatto, member and co-chair of Intellectual Property firmwide; Christopher Cuneo, patent associate; or the Mintz Levin attorney who regularly handles your legal affairs if you would like additional information on protecting your business's intellectual property. Additionally, you may e-mail the firm at IP@mintz.com to register to receive periodic updates, alerts and in-depth analyses of important intellectual property issues.

Advisory
February 2002

BIOINFORMATICS

Business and Legal Aspects of Open Source Software for Bioinformatics

*by James G. Gatto**

"Open Source promotes software reliability and quality by supporting independent peer review and rapid evolution of source code" (Source: Open Source Initiative)

The Open Source movement has gained significant momentum, and the pace is accelerating. AOL is rumored to be in talks to buy Red Hat, the largest distributor of the Open Source operating system known as Linux. Sony has announced an Open Source operating system for its Play Station platform. IBM, Sun Microsystems and other IT heavyweights have made significant Open Source announcements. The bioinformatics field has seen its share of Open Source software as well. Some of the major initiatives include BioPerl, Biopython, Biodas, BioJava, BioCORBA, BioLisp, BioRuby, Biok, EMBOSS, ENSEMBL and OpenBSA (see chart for description of what these programs do).

The goals of the Open Source movement are achieved primarily by:

- requiring distribution of Source Code including modifications, be made available to each licensee to whom object code is distributed – this enables many developers to modify software; and
- permitting the creation, use and distribution of modified and derivative works under the same terms as the license applicable to the original code.

Some Open Source licenses further prohibit use of proprietary licenses (an attempt to avoid patent issues).

Many software developers favor Open Source. While Open Source licenses vary widely, a common element is that distribution of Open Source software requires delivery of both the object code *and the source code.* Object code is the computer readable version of the code. Source code is the human readable version of the code that programmers use to write and modify code. Most commercial software is delivered only in object code form. Such code is not easily read and modified by programmers.

Some of the advantages of Open Source software include the ability for many developers to modify the code. This results in several benefits, from "peer review" for reliability and error checking purposes to rapid customization and modification of the software. On the other hand, Open Source software presents a number of potential risks, legal liabilities and other problems. The use of Open Source code also can adversely impact an organization's ability to reap the reward of its intellectual property and adversely affect its rights in other proprietary software.

This advisory provides an introduction to Open Source, addresses some benefits and drawbacks of Open Source licenses, from both a legal and business perspective, describes some Open Source software used in the bioinformatics industry, and provides some perspectives on factors to consider in deciding whether to use Open Source licenses.

*James Gatto is a member and co-chair of the Intellectual Property practice of Mintz, Levin, Cohn, Ferris, Glovsky, and Popeo, P.C. He specializes in IP issues related to Open Source software and the bioinformatics industry.

Introduction to Open Source Software

Open Source Basics

Most commercial software programs are distributed in object code format only. Copyright laws preclude users from copying, modifying or redistributing the software. In contrast, Open Source licenses typically require the distribution of source code along with the object code, and typically permit others to freely copy and modify the software, provided that any modified software that is distributed, is done so under similar license terms. Data and other output generated through the use of Open Source software is not subject to the Open Source license and may be freely sold or licensed.

What Makes Software Open Source?

Software is considered "Open Source" if it is released under an Open Source license. A distinction needs to be made between original programs and modified programs. A developer that creates an original program and releases it under an Open Source license does not relinquish its ownership rights in the software absent some further act. The owner of the software is permitted to simultaneously license the program under any combination of proprietary and Open Source licenses. In contrast, a developer that receives Open Source code and modifies it, must redistribute it pursuant to the terms of the Open Source license under which the developer received the code. Some Open Source licenses permit distribution of modified software under a combination of proprietary and Open Source licenses. Other Open Source licenses prohibit the release modified code from being released under a proprietary license.

Most Open Source licenses do not *require* redistribution of the software. (Some licenses require disclosure of errors and bugs and any fixes made). In most cases you may use or modify the software freely for your own use. Merely making modifications does not obligate you to disclose the modifications or redistribute the software with the modifications. If you chose to release the modifications, the you must typically do so under the terms of the license under which you received the software in the first place. This typically requires you to release it with the source code.

Open source licenses come in many varieties. The GNU GPL license is among the most "open." This provides some of the most extreme benefits and drawbacks, depending on your perspective. Other licenses such as the BSD and Sun Community Source License try to strike a better balance. Many Open Source licenses exist and more are being created. If you intend to develop and/or distribute software under an Open Source license, one of the most important things you can do is *make sure you read and understand the license.* While this sounds obvious, failure to do so can impact you in ways you may never imagine. For example, the QT license is governed by Norwegian law and all disputes are resolved in the City Court of Oslo!

Many Open Source programs are platforms or "infrastructure" type programs. They enable collaboration or access to various databases. In some cases proprietary applications are used with Open Source platforms to process, analyze, or visualize data retrieved using the Open Source program. Applications usable with Open Source programs can remain proprietary under certain circumstances. If, however, the proprietary program is "linked" with the Open Source software, distribution can cause the proprietary application to become Open Source.

Legal Aspects of Open Source

The legal foundation for Open Source licenses rest primarily upon the pillars of copyright law. U.S. Copyright law grants a copyright owner the right to prevent others from copying, modifying or redistributing copyrighted software. Open Source licenses permit users of Open Source software to do that which they normally would be excluded from doing — copying, modifying, and redistributing the software. The licenses also ensure the source code will be available to facilitate such modifications. In exchange for these rights, Open Source users must agree that, if they modify and release the Open Source software, they will do so under the same or similar license terms. That is, they will release the modified source code and permit others to copy, modify and redistribute the software. In theory, this benefits everyone who wants to use the software because it facilitates the peer review, error correction and enhancement of the software.

Many people erroneously believe that Open Source software can not be patented. Some Open Source licenses preclude proprietary licensing (presumably including patent licenses) to users of Open Source software. This does not mean that developers can not obtain patents on novel features, functions, algorithms, and other patentable aspects of the software they develop. Under some Open Source licenses, such patents may not be enforceable against downstream users under the Open Source license. However, some Open Source licenses expressly permit patent enforcement. In all cases, if another entity develops software with those patentable features using proprietary software, the patent owner can enforce the patents against that entity. At least one company (FSMLabs) has obtained patent rights for Open Source software. FSM has a free patent license to those who use the Open Source software including those features, and a separate royalty bearing license for others. (To see the license, go to www.fsmlabs.com).

Some Benefits and Drawbacks of Open Source Licenses.

Benefits

Open Source software provides many benefits. Some of these benefits include access to source code, the freedom to modify or customize software for specific needs and redistribute the software. The broad community of developers permits peer review and enhances reliability by facilitating identification and correction of "bugs" or others errors. It is often cheaper than commercial software (in many cases it is free). Many other benefits exist.

Drawbacks

Open Source suffers from potential drawbacks. For companies that make money by selling software, some of these drawbacks include the requirement to disclose source code for software and modifications, and authorization for third-parties to create derivative works. Combining proprietary software with Open Source, even if inadvertent, can

License	Licensed code – distribute under proprietary license	Modifications – can distribute modified software under proprietary license		Combinations – can distribute combinations under proprietary license		Licensor grants express patent license for use of software		Patent license terminates upon infringement suit v. Licensor
		Object Code	Source Code	Object Code	Source Code	Licensed Code	Modifications	
GPL	No	No	No	No	No	No	No	N/A
LGPL	No	No	No	Yes [1,2,3]	Yes [1,2,3]	No	No	N/A
BSD	Object or Source	Yes	Yes	Yes	Yes	No	No	N/A
X	Object or Source	Yes	Yes	Yes	Yes	No	No	N/A
MIT	Object or Source	Yes	Yes	Yes	Yes	No	No	N/A
Apache	Object or Source	Yes	Yes	Yes	Yes	No	No	N/A
Mozilla	Object Only [1]	Yes [1]	No	Yes [1,2]	Yes [1,2]	Yes	No	Yes
IBM	Object Only [1]	Yes [1]	No	Yes [1,2]	Yes [1,2]	Yes	No	Yes
Sun	Object Only [1]	Yes [4]	No [4]	Yes [2,4]	Yes [2,4]	Yes	No	Yes
Apple	Object Only [1]	Yes [1,5]	No [6]	Yes [1,2,5]	Yes [1,2,5]	Yes [6]	Yes [6,7]	Yes
Public Domain	Yes	Yes	Yes	Yes	Yes	No	No	N/A

[1] Licensee must make available the source code for Licensed Software (including modifications) under the applicable Open Source license.

[2] The source code of any separate and independent software need not be provided.

[3] The proprietary license must permit recipients to modify the combined software for their own use and to reverse engineer the software for the purpose of debugging such modifications.

[4] The Sun Public Licenses does not permit the release of the source code under a proprietary license. However, the Sun Industry Standards License (which is very similar to the Public License) does not require the licensee to make available the source code for the modified version of the software as long as it makes available the source code for the Licensed Software.

[5] The source code for all modifications must be made publicly available under the terms of the Apple License if licensee uses the software for anything other than personal use or internal R&D.

[6] The Apple License requires that the licensee grant a corresponding license under its patents to use the Licensed Software with the licensee's modifications pursuant to the terms stated in section 2 of the Apple License.

[7] The licensor grants a license under its patents to use the Licensed Software with any modifications for personal use and internal R&D. The patent license extends to cover all commercial uses of the Licensed Software and modifications if the licensee makes the source code for all modifications publicly available under the terms of the Apple License.

subject proprietary software to Open Source license terms. If the proprietary software is your own, you may lose rights in the software. If it is someone else's software that you use under license, you may face liability for violating the terms of the license.

Use of the software occurs with various risks. Open Source software licenses expressly disclaim all warranties. The software is distributed "as is." If the software does not work, or worse, causes damage, no recourse is available. If many developers modify the software in incompatible ways, "forking" of the software can occur. This results in inconsistent versions of the software that cannot be consolidated. Because no commercial entity stands behind the software, there is often no support for the software. Additionally, a number of Open Source projects have become defunct. One example is the Bioxml project.

Another drawback is the potentially limited ability to commercialize the software and IP. Open Source licenses do not prohibit the commercialization of research efforts or exploitation of IP – but may significantly impact the value. You may obtain copyrights and patents on Open Source software. Under some Open Source licenses, you can not seek patent licenses on released software against Open Source users. Under some Open Source licenses software may be released under proprietary licenses. Even if the Open Source applicable license precludes proprietary license, you can always seek to license patents to parties that develop and sell proprietary software (not using Open Source software).

Potential Legal Liability Issues

Patent & Copyright Infringement. Open Source software licenses disclaim liability for patent and copyright infringement. A party that develops or modifies an Open Source program might not have the right to convey a license to use the software. One reason may be that the developer illegally or inadvertently used third party proprietary software in the program. Another reason may be that the developer failed to obtain ownership of the code and the related copyrights or patents ("IP Rights") from an employee or contractor that worked on the program. Either of these, or other scenarios, can lead to potential IP claims against users or subsequent modifiers of the software. Under commercial software licenses, oftentimes a user obtains a warranty or at least an indemnity against IP infringement claims.

Involuntary Plaintiff. Open Source licenses are enforced by the copyright owners. In most cases, an Open Source program contains copyrighted works of a number of parties. If you modify the source code of Open Source software, you own the copyright to those modifications. If enforcement is necessary, you may end up being an "involuntary plaintiff" (*i.e.* a party to a litigation that you do not want to join). The GPL license encourages people to assign ownership of such copyrights to the Free Software Foundation (FSF) to enable a single entity to enforce Open Source licenses. If you assign your copyrights to the FSF, you may overcome the involuntary plaintiff problem, but you may be surrendering significant IP rights.

Releasing Open Source Software. The decision to release software under an Open Source license implicates a broad range of social, economic and legal factors (only some of which are discussed above). A company or other entity can deal with Open Source software in a number of ways. Some of the options include:

- Absolute Prohibition
- Presumptive Prohibition except if specifically authorized
- Permit Certain Types of Open Source
- Presumptive Authorization unless reason to specifically prohibit
- No Prohibition

Some factors to consider in deciding on whether to use or release software under an Open Source license include the following:

- Nature of Software — is it infrastructure or an application? Is it an original work or a modification? Is there significant IP protection available?
- Potential Ability to Commercialize/ License — Does it have independent commercial value?
- Where's the Value — the software itself or output (*e.g.,* data)?
- Burdens of deciding on case-by-case basis
- Benefits to employees/company vs. Potential Liabilities
- Potential Impact on other Software — does this software incorporate proprietary software that may be adversely impacted
- Which Open Source License(s) are you dealing with
- Any contractual obligations (*e.g.* funding agreement that gives IP rights to partner) that may preclude grant of rights to others under an Open Source license

Bioinformatics Open Source Software

Project Name	Description	License Type
BioPerl	A coordinated effort to collect computational methods routinely used in bioinformatics and life science research into a set of standard CPAN-style, well-documented, and freely available Perl modules.	Bioperl code is released under the terms of the Perl Artistic License. www.bioperl.org
Biopython	The Biopython project is a distributed collaborative effort to develop Python libraries and applications for bioinformatics. Python was chosen because it provides the cleanest way to represent the relationships between the different algorithms and data types while also being understandable to a broad range of users. These relationships should be independent of the language, so a good exchange of ideas is expected between Biopython and groups like Bioperl and the OMG LSR.	Python libraries are typically available under the Python license, which places fewer restrictions on what third party developers can do with the code (but is less free, in the GNU sense) . "We'll use the Python license for now, but if we find we need code, e.g. from Bioperl, which cannot be redistributed under the Python license, we will change to the Perl model. This is possible since the Python license is "compatible" with both the GPL and the Artistic License. www.biopython.org
Biodas	The Project focuses on the development of an Open Source system for exchanging annotations on genomic sequence data. The distributed annotation system (DAS) is a client-server system in which a single client integrates information from multiple servers. It allows a single machine to gather genome annotation information from multiple distant web sites, collate the information, and display it to the user in a single view with little coordination by the various information providers	www.biodas.org
BioJava	The BioJava Project is dedicated to providing Java tools for processing biological data. This will include objects for manipulating sequences, file parsers, CORBA interoperability, access to ACeDB, dynamic programming, and simple statistical routines. The BioJava library is useful for automating those daily and mundane bioinformatics tasks. As the library matures, the BioJava libraries will provide a foundation upon which both free software and commercial packages can be developed.	BioJava is distributed under LGPL. This means that you can use the libraries without your software being forced under either the LGPL or GPL. www.biojava.org
BioCORBA	The BioCORBA Project provides an object-oriented, language neutral, platform independent method for describing and solving bioinformatic problems. BioCORBA's mission is to leverage the code of the other Bio projects in a simple and easy to use fashion. For example language neutral environment allows users to write programs using BioPython and access BioPerl modules through the CORBA server.	www.biocorba.org
Bioxml		defunct
BioLisp	BioLisp.org is a public resource supporting scientists who use Lisp to develop intelligent applications in the biological sciences. A goal is to collect and disseminate Lisp biocomputing code, and gather pointers to Lisp and other Intelligent BioComputing methods.	Most BioLisp files are public domain. www.biolisp.org
BioRuby	BioRuby project aims to implement integrated environment for Bioinformatics by using Ruby.	Ruby is free software, distributed under GPL, or an Artistic-like, less restrictive license. www.bioruby.org
Biok	Biok is a prototype of a programmable application for biologists. Its purpose is twofold: to analyze biological data such as biological DNA or protein sequences, and to support tailorability and extensions by the end user through an integrated programming environment. The central analysis tools of the system are an alignment editor built on a spreadsheet with programmable visualization features, a sequence editor, and a plot editor. A tree editor is under development. Biok encompasses a Pise interface generator and is able to run EMBOSS applications. It is an integration environment, based on user interface building features and on a dataflow model which enables to connect all these components together by formulas.	Biok is available to the users of the Pasteur Institute. www-alt.pasteur.fr/~letondal/biok/
EMBOSS	EMBOSS is a software analysis package specially developed for the needs of the molecular biology (e.g. EMBnet) user community. The software automatically handles data in a variety of formats and allows transparent retrieval of sequence data from the web. Extensive libraries are provided with the package to provide a platform to allow other scientists to develop and release software in true Open Source spirit. EMBOSS also integrates a range of currently available packages and tools for sequence analysis into a seamless whole.	The applications are released under GPL, the libraries under LGPL. www.uk.embnet.org/Software/EMBOSS)
ENSEMBL	ENSEMBL is a joint project between EMBL - EBI and the Sanger Institute to develop a software system which produces and maintains automatic annotation on eukaryotic genomes. It enables Identification of 90% of known human genes in the genome sequence, prediction of 10,000 additional genes, all with supporting evidence, and connections to other resources worldwide, leveraging many public genomic databases and tools. Provides a web-based genome browser.	ENSEMBL permits the "download of all data code, free, without any constraints." www.ensembl.org