



Open Source and the Secrets of **Commando Due Diligence**

Heather Meeker, Shareholder,
Greenberg Traurig, LLP

March 26, 2008

Open Source Representation Circa 1995

The Company Intellectual Property contains no software source code that is covered by a so-called “open source,” “copyleft” or “general public” license. For purposes of this paragraph, “open source” is any software that is made generally publicly available in source code form.

- Definition of “open source” above is overbroad – includes non-compiled software
- Does not allow for the possibility that there is open source code in the target’s products

Open Source Representation Circa 1997

Except as set forth in the Disclosure Schedule, The Company Intellectual Property contains no software source code that is covered by a so-called “open source” license. For purposes of this paragraph, “open source” is any software that is made generally publicly available under licensed approved by the Open Source Initiative.

- Closer, but the definition of “open source” above is too narrow. The Open Source Initiative does not approve all licenses that fit the definition, only licenses that are submitted to them for approval, and even then may reject duplicative licenses. Open-Source Referees Change the Rules, Eweek, April 7, 2005 <http://www.eweeek.com/article2/0,1895,173791,00.asp>.

Open Source Representation Circa 1997

(continued)

- There are many variations of licenses such as BSD, MIT, and Apache that are not on the official list of approved licenses
- Fails to make this distinction between code used in product development, and used as back end systems
- Sprang from an assumption, since empirically doubted, that the primary liability and risk associated with open source was the possibility that, due to the number of its contributors, open source was likely to infringe third party copyrights.

Open Source Representation Circa 2002: Disclosure of Facts Regarding Open Source Software

Section _____ of the Disclosure Schedule:

lists all Open Source Software that at any time prior to the Effective Date has been used in the conduct of the Business;

describes how each element of such Open Source Software has been used in the conduct of the Business;

states whether such Open Source Software has been distributed to any third party, and in the case such Open Source Software has been so distributed, describes such distribution; and

states whether such Open Source Software has been modified by or for Company.

Open Source Representation Circa 2002

(continued)

- More sophisticated approach
- Distinguishes open source merely used in a business from open source embedded in products.

Open Source Representation Circa 2005: Disclosure of Facts Regarding Copyleft Software

Section ____ of the Disclosure Schedule:

lists all Open Source Software that at any time prior to the Effective Date has been included in any Company Product;

states whether such Open Source Software has been modified by or for Company; and

for all Proprietary Software that is distributed in connection with or interoperates with any Free Software, describes the interfaces between such Proprietary Software and Free Software (such as dynamic or static linking, forking, or communications protocol interfaces), and lists the names of any related published APIs.

Open Source Representation Circa 2005: Disclosure of Facts Regarding Copyleft Software

Section ____ of the Disclosure Schedule:

lists all Open Source Software that at any time prior to the Effective Date has been included in any Company Product;

states whether such Open Source Software has been modified by or for Company; and

for all Proprietary Software that is distributed in connection with or interoperates with any Free Software, describes the interfaces between such Proprietary Software and Free Software (such as dynamic or static linking, forking, or communications protocol interfaces), and lists the names of any related published APIs.

- Distinguishes between permissive licenses and hereditary licenses like GPL, LGPL and MPL.

Open Source Representation Circa 2005: Disclosure of Facts Regarding Copyleft Software

(continued)

■ Definitions:

- » “**Open Source Software**” means any software that is generally available to the public in source code form under licenses substantially similar to those approved by the Open Source Initiative and listed at <http://www.opensource.org/licenses/>, which licenses include without limitation the GNU General Public License, the GNU Lesser Public License, the Berkeley Science Division (BSD) License, and the Apache License.
- » “**Free Software**” means any Open Source Software licensed under a license agreement that requires, as condition of being distributed or otherwise, that the source code for the Open Source Software or any derivative works thereof be made available to licensees to whom such software is distributed, including without limitation any software licensed under the GNU General Public License, the GNU Lesser General Public License, the Mozilla Public License, the Common Public License, or the Common Development and Distribution License.
- » “**Proprietary Software**” means any portion of the Company Product that has been distributed by Company in object code form only (or that is available in source code form only upon the occurrence of an event such as cessation of business or business distress, such as part of a software source code escrow).

Due Diligence Process for Open source

- Send diligence request
- Receive response
- Vet and correct information
- (Assume here that you are not using a code consultant like Black Duck/Palamida)
- Add statements regarding compliance for hereditary licenses
- Resolve any issues
-Profit!

Why is it so difficult?

- License Information is wrong
 - » Misidentification of License
 - » Missing version numbers for licenses
- Information for identified software is incomplete
 - » Missing version numbers for software
 - » Missing dual license information
- Software is not identified

Misidentification of Licenses

- Engineer who compiled the information inaccurately identified the license
- Engineer relied on the information provided by the project, but project information is ambiguous or wrong
 - » E.g. LGPL items in GPL projects
 - » E.g. broken links to GPL3 vs. GPL2
 - » E.g. Project confusion
- The only way to be sure is to download the code and look in the source files

Information for identified software is incomplete

- Missing version numbers for software
 - » Target needs to provide, but latest version is a starting assumption
- Missing version numbers for licenses
 - » Can often be solved applying common sense and Web investigation
- Missing dual license information
 - » Not necessary unless there is a compliance problem

Software not identified

- Look for “dependencies” in project statements
- Use common sense
 - » A Firefox plug-in requires Firefox
 - » Media products require media players
 - » Most web services use Apache, Java

Rules of the road: Example

License	Compliance Rules	Comments
GPL version 2	<ul style="list-style-type: none"> • No proprietary code may be linked (dynamically or statically) to GPL code. • Proprietary applications may run in user space on top of Linux, even though Linux is covered by GPL. • GPL code may interface with proprietary code through pipes, sockets, data files, or fork/exec calls. 	<ul style="list-style-type: none"> • This generally follows the Free Software Foundation's interpretations and guidance, though there are gray areas around the margins
GPL version 3	<ul style="list-style-type: none"> • Do not use 	
LGPL version 3	<ul style="list-style-type: none"> • Do not use 	
LGPL version 2.1	<ul style="list-style-type: none"> • LGPL code may not be used by proprietary code except via dynamically linked libraries. • LGPL code may not contain macros or in-line functions of over 10 lines. 	<ul style="list-style-type: none"> • Legal must confirm that license for proprietary code does not prevent modification or reverse engineering to the extent necessary to allow exercise of LGPL rights for the library code, including modification of library and re-build of product with modified library. • Legal must confirm no patent issues
GPL v.2 + Special Exception	<ul style="list-style-type: none"> • GPL + Exception code may be used if it is an unmodified library 	<ul style="list-style-type: none"> • GPL + Exception allows proprietary code to be linked, dynamically or statically. Therefore unmodified code is unlikely to violate compliance rules • .Legal must confirm no patent issues

Rules of the road: Example

(continued)

License	Compliance Rules	Comments
Mozilla 1.1	<ul style="list-style-type: none">• Proprietary code may be combined with MPL code if it is in separate source files• MPL code may be used if it is an unmodified library	<ul style="list-style-type: none">• Legal must confirm no patent issues• Note that some companies will place Mozilla 1.1 on the black list due to patent termination provisions; this may not apply to Mozilla 1.0
CDDL	<ul style="list-style-type: none">• Proprietary code may be combined with CDDL code if it is in separate source files• CDDL code may be used if it is an unmodified library	<ul style="list-style-type: none">• Legal must confirm no patent issues• Note that some companies will place CDDL on the black list due to patent termination provisions
Eclipse	<ul style="list-style-type: none">• Eclipse code may be used if it is an unmodified library	<ul style="list-style-type: none">• Legal must confirm no patent issues
CPL	<ul style="list-style-type: none">• CPL code may be used if it is an unmodified library	<ul style="list-style-type: none">• Legal must confirm no patent issues
Apache 2.0	<ul style="list-style-type: none">• OK	<ul style="list-style-type: none">• Legal must confirm no patent issues
BSD	<ul style="list-style-type: none">• OK	
MIT	<ul style="list-style-type: none">• OK	
Apache 1.1	<ul style="list-style-type: none">• OK	
Apache 1.0	Do not use	<ul style="list-style-type: none">• Note that some companies will place Apache 1.0 on the black list due to its advertising clause, since deprecated in favor of Apache 1.1

I've found a problem – now what?

■ Remove

- » Lots of code bases contain code that is not used (like “junk DNA”)

■ Reengineer

- » Only applicable for hereditary licenses
- » “Shimming” usually does not work

■ Re-license

- » Dual licensing
- » Private re-licensing
- » Version backtracking (most applicable to GPL/LGPL 3 issues)

Commando Due Diligence

Case Study 1: Commercial dual licensing

- Target was using RED5 (open source flash server licensed under LGPL2) and wanted to replace it with Adobe Blaze DS (licensed under LGPL3) for technical reasons on the eve of the acquisition
 - » Acquiror would not accept GPL3
 - » Resolution:
 - We found a commercial version of Blaze licensed under commercial terms
 - We deducted the license fees (based on current use) from the purchase price
 - Acquiror approved additional fees necessary for ramp-up after acquisition

Commando Due Diligence

Case Study 1: Commercial dual licensing

- Target was using RED5 (open source flash server licensed under LGPL2) and wanted to replace it with Adobe Blaze DS (licensed under LGPL3) for technical reasons on the eve of the acquisition
 - » Acquiror would not accept GPL3
 - » Resolution:
 - We found a commercial version of Blaze licensed under commercial terms
 - We deducted the license fees (based on current use) from the purchase price
 - Acquiror approved additional fees necessary for ramp-up after acquisition

Commando Due Diligence

(continued)

Case Study 2: Commercial Substitute

- Target was using Octave (math library licensed under GPL) statically linked
 - » Resolution:
 - We found a commercial product (MATLAB) with similar functionality and minimal license fees.
 - We required re-licensing as a closing condition.

Commando Due Diligence

(continued)

Case Study 3: Private Re-licensing

- Target was using FRIBIDI (an implementation of the Unicode Bidirectional Algorithm (bidi) licensed under GPL) and linking to proprietary code.
 - » Resolution:
 - We found the author and entered into a one-off binary license for about \$10,000
 - In this case, acquiror absorbed the fee
 - This works best for individual authors
 - Will not work for most large projects or projects with “faith-based” licensing. Note that FRIBIDI has since become a GNU project.

Commando Due Diligence

(continued)

Case Study 3: Private Re-licensing

- Target was using FRIBIDI (an implementation of the Unicode Bidirectional Algorithm (bidi) licensed under GPL) and linking to proprietary code.
 - » Resolution:
 - We found the author and entered into a one-off binary license for about \$10,000
 - In this case, acquiror absorbed the fee
 - This works best for individual authors
 - Will not work for most large projects or projects with “faith-based” licensing. Note that FRIBIDI has since become a GNU project.

CV on Open Source

- Author of *The Open Source Alternative*, John Wiley & Sons, 2008.
- Advised Autodesk on OSGEO Foundation formation and code release
- Advised Mozilla on marketing agreement Firefox search defaults
- Drafted Firefox EULA for Mozilla Foundation
- Advisor for open source issues to Yahoo, Autodesk, Avaya, Amazon.com, Ebay, Vuze, Serena, TIBCO, LSI Logic, Pace Micro, Insightful, Sony.
- Prepared trademark and patent policies, contribution policies, and licensing strategies for Active Endpoints, Alfresco Software, Cobia (StillSecure), Jahshaka, Boingo, Centeris, Digium, Second Life.

CV on Open Source

(continued)

- Advised Yahoo! on Zimbra, Right Media, and other acquisitions.
- Advised on open source and intellectual property matters in Network Associates' acquisitions of Intruvert Networks, Entercept Security Technologies, Deersoft, and Traxess, 2002-2003.
- Counseled Mozilla Foundation, Open Source Applications Foundation, GNOME Foundation, Python Software Foundation
- Member of Open Bar Advisory Board
- Co-Chair of Open Source committee for ABA
- Advised Lucas Arts on initiating OpenEXR open source code release
- Advisory Member for ALI project on the Law of Software Contracts

Thank you for your time.

Heather J. Meeker
Greenberg Traurig, LLP
Silicon Valley Office
1900 University Ave, 5th floor
East Palo Alto, CA 94303
650-289-7825