Advanced Licensing Agreements 2017

Volume One

Co-Chairs
Marcelo Halpern
Ira Jay Levy
Joseph Yang

To order this book, call (800) 260-4PLI or fax us at (800) 321-0093. Ask our Customer Service Department for PLI Order Number 185480, Dept. BAV5.

Practising Law Institute
1177 Avenue of the Americas
New York, New York 10036
24

Open Source Issues and Opportunities
(PowerPoint slides)

David G. Rickerby

Boston Technology Law, PLLC

If you find this article helpful, you can learn more about the subject by going to www.pli.edu to view the on demand program or segment for which it was written.
Open Source Issues and Opportunities

Practicing Law Institute
Advanced Licensing Agreements 2017

May 12th 2017
10:45 AM - 12:15 PM
David G. Rickerby
Overview

- Introduction to Open Source
- Enforced Sharing
- Managing Open Source
What is Open Source?

- “Open” “Source” – Any software licensing model that makes the source available to copy, distribute, modify, etc.
What is Source Code?
- The human readable version of the code.
- Exposes trade secrets, interfaces, and logic.

Private Function CleanUpLine(ByVal sLine As String) As String
Dim iQuoteCount As Long
Dim iCount As Long
Dim sChar As String
Dim sPrevChar As String

' Starts with Rem it is a comment
sLine = Trim(sLine)
If Left(sLine, 3) = "Rem" Then
    CleanUpLine = ""
    Exit Function
End If

' Starts with ' it is a comment
If Left(sLine, 1) = "'" Then
    CleanUpLine = ""
    Exit Function
End If

' Contains ' may end in a comment, so best if it is a comment or in the body of a string
If InStr(sLine, "'" ') > 0 Then
    sPrevChar = ""
    iQuoteCount = 0

    For iCount = 1 To Len(sLine)
        sChar = Mid(sLine, iCount, 1)
        If sChar = "" And sPrevChar = "" Then
            iQuoteCount = iQuoteCount + 1
        End If
        If iQuoteCount Mod 2 = 0 Then
            sLine = Trim(Left(sLine, iCount - 1))
            Exit For
        Else
            sLine = Trim(Right(sLine, Len(sLine) - iCount + 1))
        End If
    Next iCount
    If sChar = "" And sPrevChar = "" Then
        Exit Function
    End If
End Function
As opposed to Object Code...
Open Source is Big Business

ANDROID - Declared license: Apache 2.0
~185 components
~19 different OSS licenses - most reciprocal
Solving Problems in Many Industries

Healthcare
OSEHRA
Open Source Electronic Health Record Act

Financial Services
THE LODESTONE FOUNDATION

Everything
ALLSEEN ALLIANCE

Mobile

Automotive
GENIVI

BostonTechLaw
So, what’s the big deal? Why isn’t this just like a commercial license?

In many ways they are the same:

- Both commercial and open source licenses are based on ownership of intellectual property.
- Both grant certain rights and retain others.
- Both are governed by the same laws.
- Both may include provisions which may be incompatible with the other type of license, and, indeed with other licenses of the same type.
But...

Open Source Licenses:
- Tend to have different goals
- Are usually written by and for developers not lawyers
- Encourage uncontrolled combination and reuse of the IP
- Form a contract in a different way than most commercial licenses (in fact some argue they don’t form a contract at all – merely act as a permission)

AND

- Some open source licenses impose sharing obligations on users
Two Basic Schools of Open Source

- **FOSS (Free and Open Source Software)**
  - Requires licensor to make improvements or enhancements available under similar terms
  - Primary example is the GPL: Licensee must distribute “work based on the program” and cause such works to be licensed ... under the terms of the GPL

- **Academic/Permissive**
  - Modifications/enhancements may remain proprietary
  - Distribution in source code or object code permitted provided copyright notice & liability disclaimer are included and contributors’ names are not used to endorse products
  - Primary examples: Berkeley Software Distribution (BSD), Apache Software License
Free Software Definition
from the Free Software Foundation

“Free software” is a matter of liberty, not price. To understand the concept, you should think of “free” as in “free speech,” not as in “free beer.”

Free software is a matter of the users' freedom to run, copy, distribute, study, change and improve the software. More precisely, it means that the program’s users have the four essential freedoms:

• The freedom to run the program, for any purpose (freedom 0).

• The freedom to study how the program works, and change it to make it do what you wish (freedom 1). Access to the source code is a precondition for this.

• The freedom to redistribute copies so you can help your neighbor (freedom 2).

• The freedom to distribute copies of your modified versions to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.

Principles of Open Source Licensing
from the Open Source Initiative

1. Free [Unrestricted] Redistribution
2. Program must include Source Code and must allow distribution in source code as well as compiled form.
3. Must Allow Modifications and Derived Works
4. Integrity of the Author's Source Code
5. No Discrimination Against Persons or Groups
6. No Discrimination Against Fields of Endeavor
7. Distribution of License – no additional license can be required of others who redistribute the program
8. License Must Not Be Specific to a Product
9. License Must Not Restrict Other Software
10. License Must Be Technology-Neutral – not predicated on any individual technology

from http://www.opensource.org/docs/osd
## 78 Current OSI Approved Licenses (www.opensource.org)

<table>
<thead>
<tr>
<th>No.</th>
<th>License Name</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Academic Free License 3.0 (AFL-3.0)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Afferro General Public License 3.0 (AGPL-3.0)</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Adaptive Public License (APL-1.0)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Apache License 2.0</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Apple Public Source License</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Artistic license 2.0</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Attribution Assurance License (AAL)</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>BSD 3-Clause &quot;New&quot; or &quot;Revised&quot; License (BSD-3-Clause)</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>BSD 2-Clause &quot;Simplified&quot; or &quot;FreeBSD&quot; License (BSD-2-Clause)</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Boost Software License (BSL-1.0)</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>CeCILL License 2.1 (CECILL-2.1)</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Computer Associates Trusted Open Source License 1.1 (CATOSL-1.1)</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Common Development and Distribution License 1.0 (CDDL-1.0)</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Common Public Attribution License 1.0 (CPAL-1.0)</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>CUA Office Public License Version 1.0</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>EU DataGrid Software License</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Eclipse Public License 1.0 (EPL-1.0)</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>eCos License version 2.0</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>Educational Community License, Version 2.0 (ECL-2.0)</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>Eiffel Forum License V2.0 (EFL-2.0)</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>Entessa Public License (Entessa)</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>European Union Public License, Version 1.1 (EUPL-1.1)</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>Fair License (Fair)</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>Framework License (Framework-1.0)</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>Free Public License 1.0.0</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>GNU Affero General Public License v3 (AGPL-3.0)</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>GNU General Public License version 2.0 (GPL-2.0)</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>GNU General Public License version 3.0 (GPL-3.0)</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>GNU Library or &quot;Lesser&quot; General Public License version 2.1 (LGPL-2.1)</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>GNU Library or &quot;Lesser&quot; General Public License version 3.0 (LGPL-3.0)</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>Historical Permission Notice and Disclaimer (HPND)</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>IBM Public License 1.0 (IPL-1.0)</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>IPA Font License (IPA)</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>ISC License (ISC)</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>LaTeX Project Public License 1.3c</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>License Libre du Quebec – Permissive v.1.1</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>License Libre du Quebec – Reciprocite</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>License Libre du Quebec – Reciproite Forte</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>Lucent Public License Version 1.02</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>Microsoft License (MirOS)</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>Microsoft Public License (MS-PL)</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>Microsoft Reciprocal License (MS-RL)</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>MIT license (MIT)</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>Motosoto License (Motosoto)</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>Mozilla Public License 2.0 (MPL-2.0)</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>Multics License (Multics)</td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>NASA Open Source Agreement 1.3</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>NTP License (NTP)</td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>Naumen Public License (Naumen)</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>Nethack General Public License</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>Nokia Open Source License (Nokia)</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>Non-Profit Open Software License 3.0</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>OCLC Research Public License 2.0</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>Open Group Test Suite License</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>Open Software License 3.0 (OSL-3.0)</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>OSET Public License version 2.1</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>PHP License 3.0 (PHP-3.0)</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>PostgreSQL License</td>
<td></td>
</tr>
<tr>
<td>59</td>
<td>Python License (Python-2.0)</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>Q Public License (QPL-1.0)</td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>RealNetworks Public Source License version 1.0 (RPSL-1.0)</td>
<td></td>
</tr>
<tr>
<td>62</td>
<td>Reciprocal Public License 1.5</td>
<td></td>
</tr>
<tr>
<td>63</td>
<td>Ricoh Source Code Public License</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>Simple Public License 2.0 (SimPL-2.0)</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>Sleepycat License (Sleepycat)</td>
<td></td>
</tr>
<tr>
<td>66</td>
<td>Sun Public License 1.0 (SPL-1.0)</td>
<td></td>
</tr>
<tr>
<td>67</td>
<td>Sybase Open Watcom Public License 1.0 (Watcom-1.0)</td>
<td></td>
</tr>
<tr>
<td>68</td>
<td>University of Illinois/NCsa Open Source License (NCSA)</td>
<td></td>
</tr>
<tr>
<td>69</td>
<td>Universal Permissive License (UPL)</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>Vovida Software License v. 1.0</td>
<td></td>
</tr>
<tr>
<td>71</td>
<td>W3C License (W3C)</td>
<td></td>
</tr>
<tr>
<td>72</td>
<td>wxWindows Library License (WXwindows)</td>
<td></td>
</tr>
<tr>
<td>73</td>
<td>X.Net License (Xnet)</td>
<td></td>
</tr>
<tr>
<td>74</td>
<td>Zero Clause BSD License</td>
<td></td>
</tr>
<tr>
<td>75</td>
<td>zlib/libpng license (Zlib)</td>
<td></td>
</tr>
</tbody>
</table>
## 10 Most Common Open Source Licenses

<table>
<thead>
<tr>
<th>Rank</th>
<th>License</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MIT License</td>
<td>28%</td>
</tr>
<tr>
<td>2</td>
<td>GNU General Public License (GPL) 2.0</td>
<td>20%</td>
</tr>
<tr>
<td>3</td>
<td>Apache License 2.0</td>
<td>16%</td>
</tr>
<tr>
<td>4</td>
<td>GNU General Public License (GPL) 3.0</td>
<td>8%</td>
</tr>
<tr>
<td>5</td>
<td>BSD License 2.0 (3-clause, New or Revised) License</td>
<td>6%</td>
</tr>
<tr>
<td>6</td>
<td>Artistic License (Perl)</td>
<td>4%</td>
</tr>
<tr>
<td>7</td>
<td>GNU Lesser General Public License (LGPL) 2.1</td>
<td>4%</td>
</tr>
<tr>
<td>8</td>
<td>ISC License</td>
<td>4%</td>
</tr>
<tr>
<td>9</td>
<td>GNU Lesser General Public License (LGPL) 3.0</td>
<td>2%</td>
</tr>
<tr>
<td>10</td>
<td>Microsoft Public License</td>
<td>2%</td>
</tr>
</tbody>
</table>

Note: The table above lists the top licenses that are used in open source projects ranked by number of projects using the license, according to the Black Duck Software KnowledgeBase. This data was pulled on October 20, 2016.
An oversimplified summary:

- FOSS Licenses (GPL2, GPL3, Affero GPL, LGPL) are going to require you to make the source code available under the same license for the open source program and for any work you distribute which is based on the program.

- Many other Open Source licenses (Mozilla, CPL) are going to require you to make available modifications you make to the open source program, but not works which interface with it.

- A few open source licenses (Apache, BSD) are going to let you do pretty much whatever you want with the code as long as you give appropriate credit and disclaim all warranties and liability (there is no such thing as a free lunch).
Enforced Sharing: GPL v2

- “[T]he intent is to exercise the right to control the distribution of derivative or collective works based on the Program.”
- “You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.”

http://www.gnu.org/licenses/gpl.html
Enforced Sharing: LGPL

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

http://www.gnu.org/licenses/lgpl.html
Enforced Sharing: LGPL

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work ...

b) Use a suitable shared library mechanism for linking with the Library. ...

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution. ...

http://www.gnu.org/licenses/lgpl.html
Enforced Sharing: GPL v3

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License...

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.
Enforced Sharing: What is a Derivative Work?

The Copyright Act doesn’t really help for Software:

- §101, A “derivative work” is a work based upon one or more preexisting works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which a work may be recast, transformed, or adapted. A work consisting of editorial revisions, annotations, elaborations, or other modifications which, as a whole, represent an original work of authorship, is a “derivative work.”

- §102(b), In no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work.
Enforced Sharing: What is a Derivative Work?

- Abstraction, Filtration, Comparison Test
  - 2nd, 5th, 10th and 11th Circuit, Computer Associates Intl., Inc. v. Altai, Inc., 982 F.2d 693 (2nd Cir. 1992);
    - Abstract from the original program its constituent structural parts.
    - Filter unprotected portions (incorporated ideas, expression necessary to those ideas, and public domain elements).
    - Compare any and all remaining creative expression with structure of 2nd program to determine whether derivative.
  - 1st Circuit has a modified version that applies the test only after filtering out § 102(b) unprotected elements – particularly methods of operation or control. (Lotus Development Corp. v. Borland Int’l., Inc., 49 F.3d 807 (1st Cir. 1995)).
Enforced Sharing: What is a Derivative Work?

Analytic Dissection Test

- 9th Circuit has adopted the analytic dissection test (Apple Computer, Inc. v. Microsoft Corp., 35 F.3d 1435 (9th Cir. 1994))

- Are there substantial similarities in both the ideas and expressions of the works? Are the similar features (if any) protected by copyright?

- “Thin” protection is given to non-copyrightable facts or ideas that are protectable because of how they are combined/presented. “Broad” protection is given to copyrightable expression.

- Depending on the degree of protection, court sets standard for comparison of works as a whole to determine if derivative. “Thin” protection requires virtually identical; “broad” protection requires only a “substantial similarity.”
Enforced Sharing: What is a Derivative Work?

Oracle v. Google – protection of APIs

- In 2012, District Court decided that the command structure and taxonomy of the APIs were not protectable under copyright law.

- On May 9, 2014, the Federal Circuit partially reversed the district court ruling, ruling in Oracle's favor on the copyrightability issue, and remanded the issue of fair use to the district court.

- A petition for certiorari was denied by the United States Supreme Court on June 29, 2015.

- A second trial began on May 9, 2016 – Oracle requested 8.8B in damages.

- The trial jury sided in favor of Google, ruling the action to be fair use. (May 29, 2016).
Enforced Sharing: What is a Derivative Work?

> AN OVERSIMPLIFICATION OF A COMPLEX CASE

- Google wrote its own version of Java for Android using the same “taxonomy of all the names of methods, classes, interfaces, and packages” as Oracle’s Java.

- Question as to copyrightability of a taxonomy/API. Argument that this element of software is purely functional and necessary for technology systems to speak to one another.

- Federal Circuit overruled District Court finding enough creativity for copyright purposes, but sent case back for retrial on the question of Fair Use.

- Google won the retrial ruling the action to be fair use.

- Broad implications for use open source. Does interoperability make everything derivative?
Keys to Open Source

- The key to any open source analysis is the same:
  - Know and understand the facts
  - Read the license
  - Know the norms of the community from which you are taking the source
  - Play nice
GPL v3, LGPL v3, AGPL

- Other commercial concerns:
  - Anti-DRM
    - “No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.”
  - Anti-tivoization
    - If you convey an object code work ... in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source ... must be accompanied by the Installation Information.
  - Patent License Grant /Patent Non-Assertion as License Condition
  - AGPL Compatibility
Focus is Compliance

- Companies are adopting open source policies and processes:
  - To alleviate litigation concerns
  - As an important M&A Issue
  - To comply with Customer diligence
  - More participation in open source communities: what are your employees contributing?
Elements for an Open Source Program?

- Published Policy
- Open Source Process Owner
- Approval Processes
- Monitoring & Tracking Process
- Obligation Verification Process

And...

As much as I’d like to sell you my services, this is OPEN SOURCE, so check out http://www.linuxfoundation.org/programs/legal/compliance for sample policies.
Be Ready to Comply with the Letter of the Requirements

  - Monitor Software Acquisitions – ignorance is not an excuse.
  - Track you changes and releases – you need to be able to provide the exact code that is shipping.
  - Don’t have a “Build Guru” – users need to be able to successfully build your code. You need to provide the scripts and other materials that are necessary to build and install.
Common Mistakes

- Legalese: make it understandable
- Specific policy that ignores other issues
- Policy too strict so VOA: Violated on Arrival
- Does not allow for edge cases
- Does not provide for modification to meet changes in business model/products
Summary

- Treat the management of open source software as an integrated, cross-functional business process
- Establish policies, define the process and process owners
- Phase the deployment to yield near-term results
- Technology platforms can automate the process, enhance cross-functional collaboration and ensure validation
Questions?

David Rickerby, Esq.
David@bostontechlaw.com
617-848-2616