Open Source Software:
What Business Lawyers, Entrepreneurs and IT Professionals Should Know

Stuart R. Hemphill
Partner
Minneapolis
P: (612) 340-2734
F: (612) 340-8856
hemphill.stuart@dorsey.com

Source vs. Object

Source Code	Object Code

Programmer readable statements in a computer language, such as C, C++, Cobol, Fortran, Java, Perl, PHP
 	
// Create a button and add it to the applet. // Also, set the button's colors clear_button = new Button("Clear");
clear_button.setForeground(Color.black);
clear_button.setBackground(Color.lightGray);
this.add(clear_button);
	Machine readable Binary:
	
	0000101000010001010
	1100010100000010100
	0001001010010001011
	
	Or Hexadecimal
	
	3F7A
	(translates to the following binary number:
	0011 1111 0111 1010)

History of Open Source Software

- Term coined in February 1998 by Silicon Valley insiders in anticipation of Netscape's announcement that it would release the source code for its browser software
- This meant software coders could understand the browser's working details and potentially modify them
- 1998 was a momentous time for open source movement given mainstream adoption of internet
- But concept significantly pre-dates coining of term

Free Software Foundation

- Free Software Foundation (FSF), created in 1983 by Richard Stallman of MIT with goal of developing free version of UNIX operating system; everyone could share and change this version
- According to FSF, "'Free software' is a matter of liberty, not price … think of 'free' as in 'free speech,' not as in 'free beer.'"
- Stallman wrote a license leveraging copyright in base code and intended to keep derivatives of base software "free" by requiring source code disclosure
- Non-negotiable terms; accept by use
- Called, "copyleft", the concept encouraged copyright owners to require all derivative works authored by licensed users of free version also to be "free", i.e., licensed under the same "free" terms as the base software

Free Software Foundation and GPL

- FSF's "General Public License" (GPL) model became known as GNU system (GNU is recursive acronym for "GNU's Not UNIX")
  - (GPL license is recursive as to each layer of derivative work)
- For developers/sellers of proprietary software (most vendors) "copyleft" meant loss of proprietary right
- Culture war ensued: academic idealists vs. software sellers?

Open Source Initiative

- In contrast to FSF, 1998 group, which later became Open Source Initiative (OSI), saw opportunity to demonstrate advantages of open development to corporate world
- Whereas other movements focus on moral legitimacy of “free” software, OSI focuses on practical advantages of open development process
- Basic idea: when developers can freely read, modify, fix and redistribute source code, software can evolve more quickly and robustly than through traditional proprietary model
- GPL as preferred but not only model license

Proprietary License Model vs. Open Source License Model

Proprietary	Open
Licensor distributes object code; source code is trade secret	Licensor distributes source and object code
Little or no licensee right or ability to modify software	Modification by licensee permitted and enabled (source access)
High license fees; additional charge for support	No or low license fees; additional charge for available support
Limitations on use per license	No limitations on use
No or limited sublicensing	Broad sublicensing/distribution
Warranty of conformance to specification and/or IP indemnification (often)	No warranties or IP indemnification; but vendors may provide special indemnification for a fee
Attribution via traditional trademarks	Attribution as author reputation builder

Types of Open Source Licenses (2014)

Rank	License	%
1.	GNU General Public License (GPL) 2.0	33%
2.	Apache License 2.0	13%
3.	GNU General Public License (GPL) 3.0	12%
4.	MIT License	11%
5.	BSD License 2.0 (3-clause, New or Revised) License	7%
6.	Artistic License (Perl)	6%
7.	GNU Lesser General Public License (LGPL) 2.1	6%
8.	GNU Lesser General Public License (LGPL) 3.0	3%
9.	Eclipse Public License (EPL)	2%
10.	Code Project Open 1.02 License	1%
11.	Microsoft Public License	1%
12.	Mozilla Public License (MPL) 1.1	1%
13.	Common Development and Distribution License (CDDL)	< 1%
14.	BSD 2-clause "Simplified" or "FreeBSD" License	< 1%
15.	Common Public License (CPL)	< 1%
16.	zlib/libpng License	< 1%
17.	Academic Free License	< 1%
18.	GNU Affero GPL v3	< 1%
19.	Microsoft Reciprocal License (Ms-RL)	< 1%
20.	Open Software License (OSL)	< 1%

Source: http://www.blackducksoftware.com/resources/data/top-20-open-source-licenses

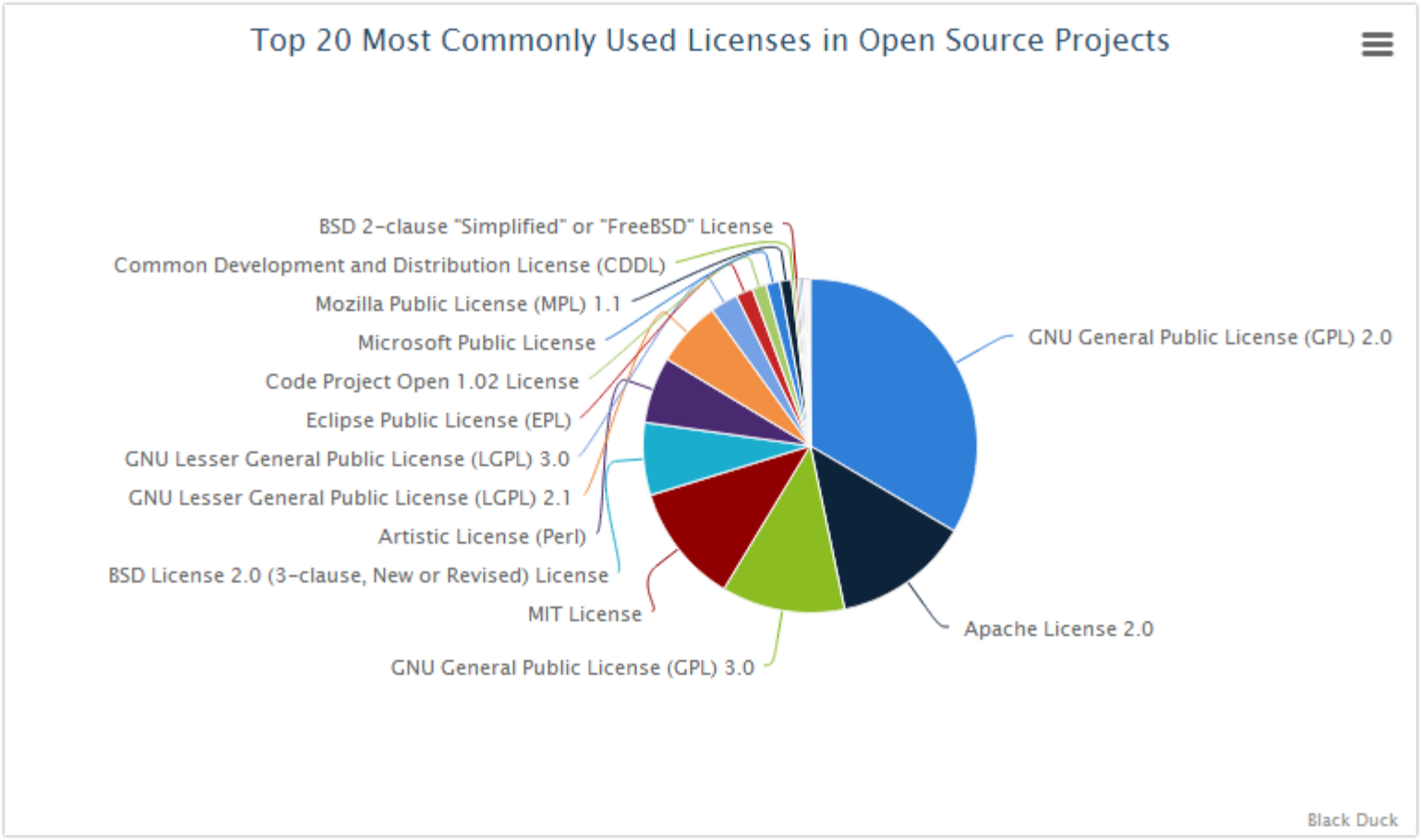
Types of Open Source Licenses (2016)

Please click on an open source license name for more information.

Rank	License	%
1	MIT License	26%
2	GNU General Public License (GPL) 2.0	21%
3	Apache License 2.0	16%
4	GNU General Public License (GPL) 3.0	9%
5	BSD License 2.0 (3-clause, New or Revised) License	6%
6	GNU Lesser General Public License (LGPL) 2.1	4%
7	Artistic License (Perl)	4%
8	GNU Lesser General Public License (LGPL) 3.0	2%
9	ISC License	2%
10	Microsoft Public License	2%
11	Eclipse Public License (EPL)	2%
12	Code Project Open License 1.02	1%
13	Mozilla Public License (MPL) 1.1	< 1%
14	Simplified BSD License (BSD)	< 1%
15	Common Development and Distribution License (CDDL)	< 1%
16	GNU Affero General Public License v3 or later	< 1%
17	Microsoft Reciprocal License	< 1%
18	Sun GPL With Classpath Exception v2.0	< 1%
19	DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE	< 1%
20	CDDL-1.1	< 1%

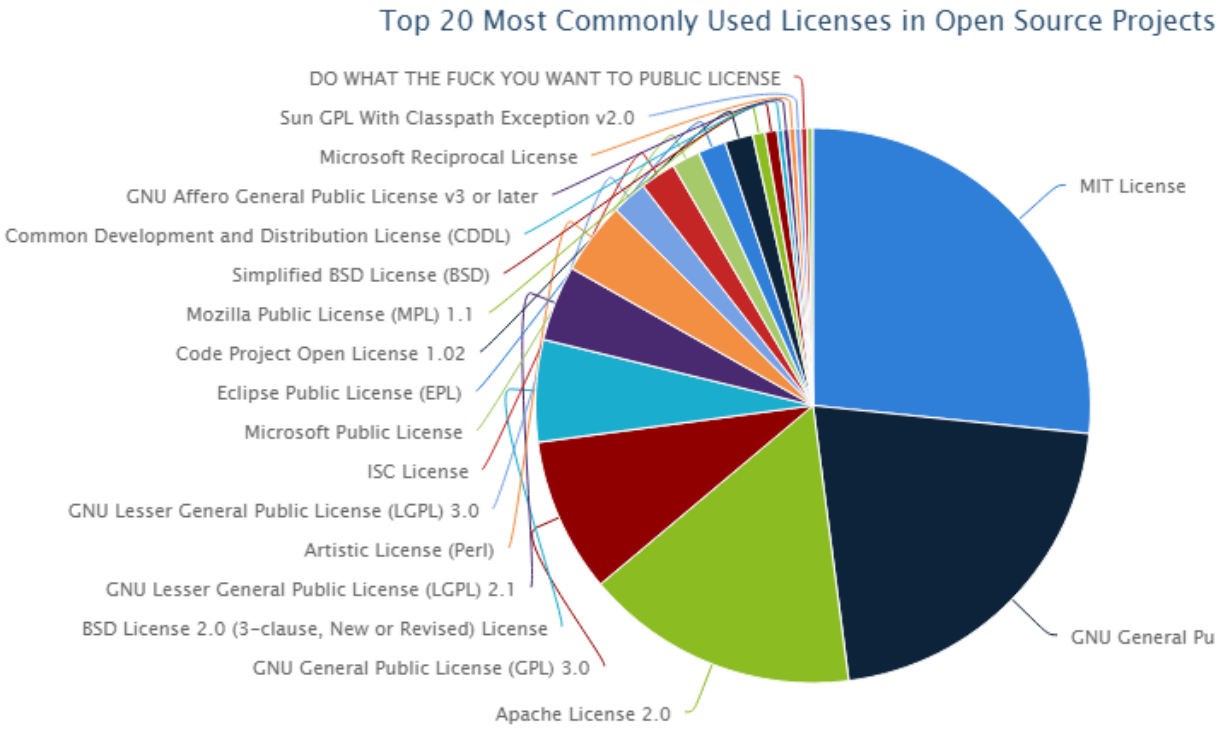
Source: http://www.blackducksoftware.com/resources/data/top-20-open-source-licenses

Types of Open Source Licenses (2014)



Source: http://www.blackducksoftware.com/resources/data/top-20-open-source-licenses

Types of Open Source Licenses (2016)



Source: http://www.blackducksoftware.com/resources/data/top-20-open-source-licenses

Main Categories of Open Source Licenses

- Full Copyleft: derivatives, defined broadly, must be licensed according to same terms that govern original (GPL, Affero)
- Limited Copyleft: only limited types of derivatives must be licensed according to same terms that govern original (Mozilla, CDDL, LGPL)
- Attribution: generally only require that credit be given to contributors (Apache, BSD, MIT)

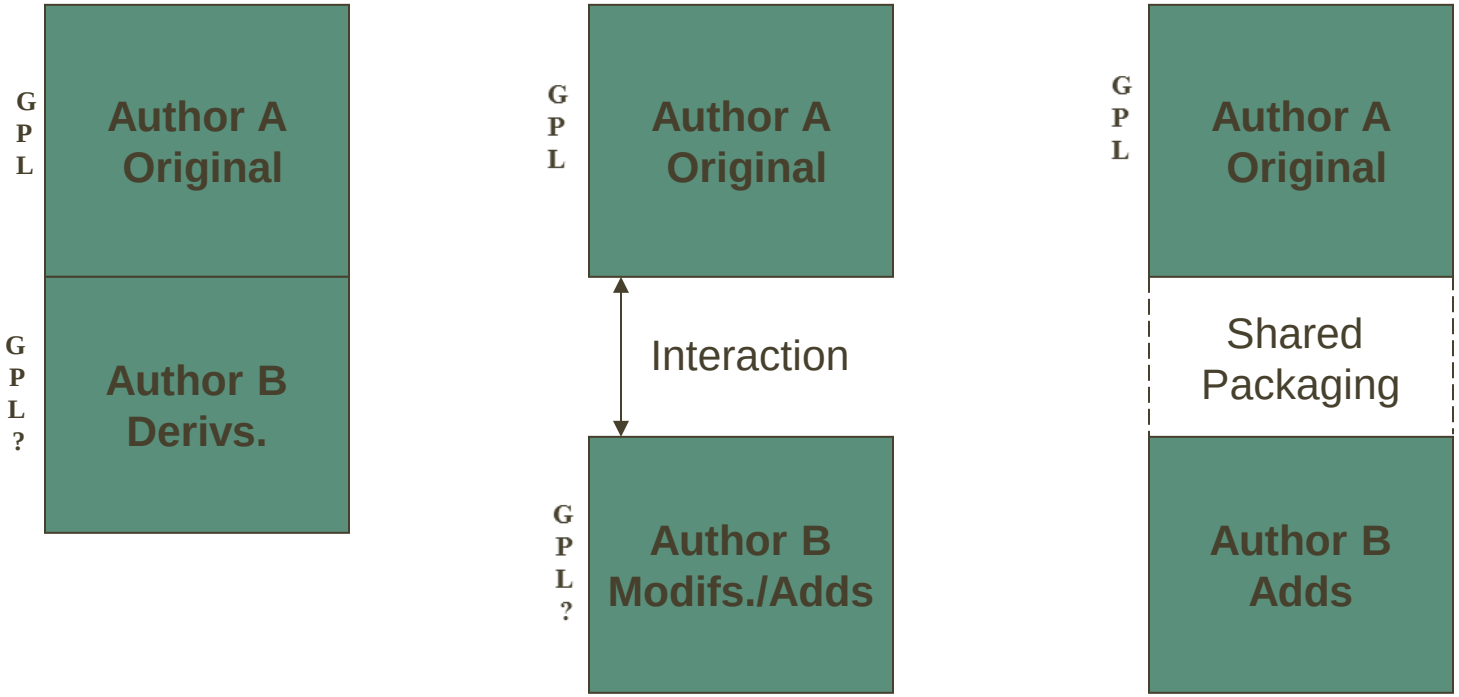
Copyleft Open Source Licenses

- In general, copyleft licenses require works that copy from, are derived from, or combined with the original work to be distributed under the same terms as the original work (sometimes referred to as “viral effect,” “infection” or “taint”)
- Forced disclosure of source code for proprietary software is main concern
- Reach of copyleft varies from license to license, and by interpretation
- In general, internal use does not trigger viral affect

Triggers for Requirement to Disclose Source Code

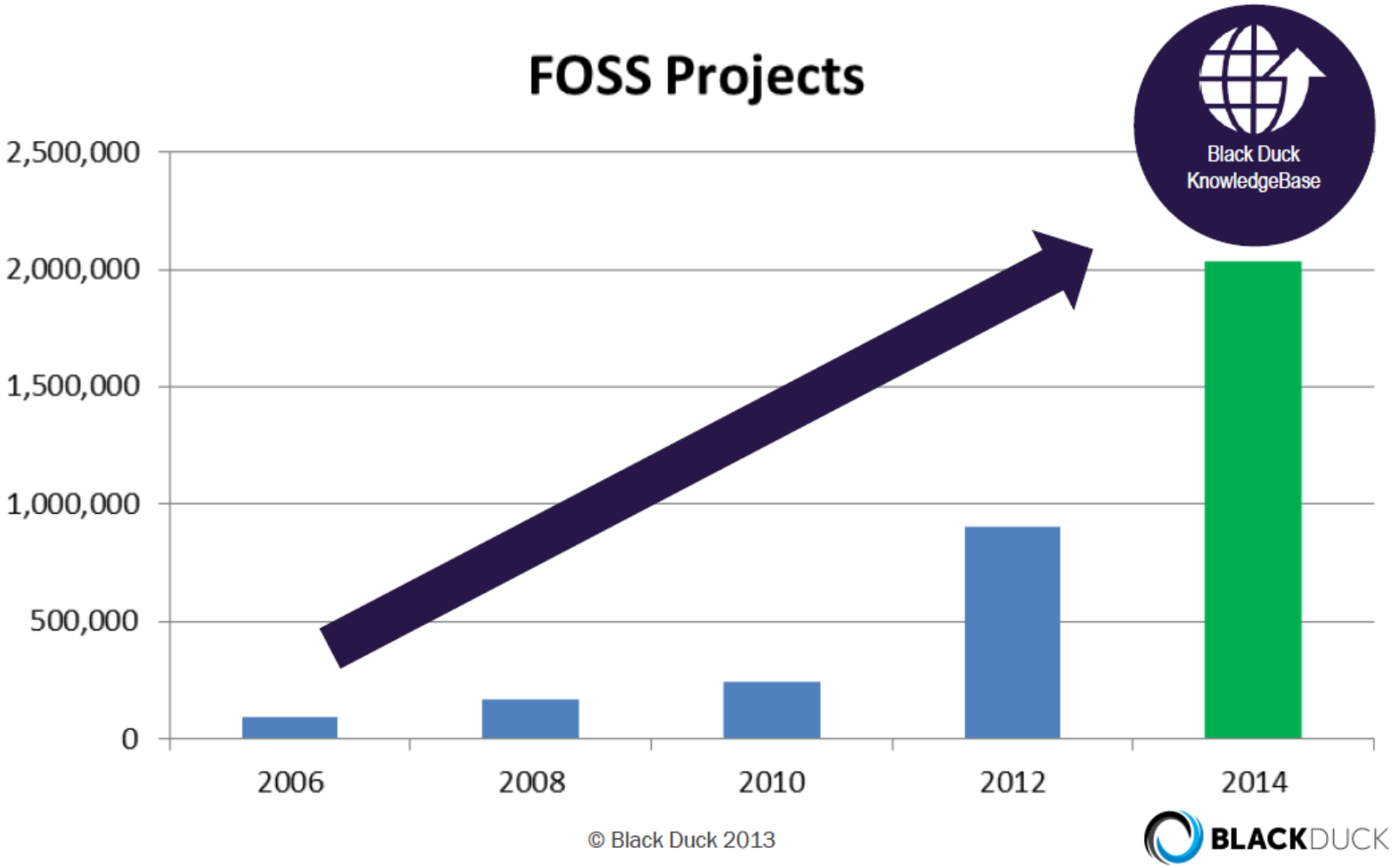
- GPL: distribution to third party (e.g., on disc, for download and local installation) requires that source code be made available; internal use or providing use of object code on server (e.g., SaaS) does not trigger disclosure requirement
- Affero: if users access object code via public server (e.g., SaaS), source code must also be made similarly accessible
  - MongoDB (AGPL for server, but Apache for drivers)
  - SugarCRM Community Edition (up to v6.5; AGPLv3)

“Capture” of Modified, Derivative or Linked Work (MDLW)



Interaction: Intertwined functions; calls, links (static, dynamic); pure add-on (see debate at http://en.wikipedia.org/wiki/GNU_General_Public_License)

Proliferation of Open Source Software



Advantages of Using Open Source

- Efficiency: no need to reinvent the wheel
- Freedom: access to source code for customization
- Cost: typically free (as in “free beer”)
- Quality: many open source offerings regarded as best in field

Risks of Using Open Source

- Forced Disclosure: GPL, but not all licenses
- No warranty: authors cannot reasonably warrant
- Usually No Support: typically author does not support; possible community support?
- No Road Map: path of development usually not available
- Security: code of unknown origin and security design

Open Source Business Models

- Secondary Commercial Services: rather than sell software, company provides support and integration services for a fee (Red Hat)
- Loss Leader/Support of Secondary Products: company opens code to jumpstart infant market, break into existing market with entrenched closed-source competitors or position itself prominently in market (Google's Android code)
- Resource Pooling: companies cooperate to develop software used by all that may be too costly or difficult for any one company to develop alone (OpenStack cloud operating system; participants include Intel, Rackspace, HP, IBM, Cisco, VMware, Oracle, Comcast, AT&T)
- Credentials: contributions to open source projects can augment developer's reputation (Github rankings)

Patent Implications of Open Source Licenses

- If developers bring in Open Source as component or tool, software that is a modification or derivative of or linked to it (MDLW) may become subject to copyleft requirements stated in the OS license
- Per GPL, software must be disclosed to and licensed to others under the same license terms
- "You must cause any work that you distribute or publish, . . . to be licensed as a whole at no charge to all third parties under the terms of this License."
- For others to use and distribute your software under a license, that license must be implied to include rights in any patents you control that cover making, using, selling, importing the software

Patent License

- Most OS licenses speak only of permitted acts and do not identify specific IP that is licensed
- E.g., BSD License grant:
- "Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: . . . "
- Implicit patent license from license to use software; but what is scope? One commentator (Lawrence Rosen)
  - *An open source license must grant enough patent rights to allow the licensee to make, use, sell, offer for sale, or import the open source work as distributed by its licensor. Any additional license rights for derivative works or other uses are at the option of the licensor*.

Open Source Patent Implications

- With any open source license attaching to your MDLW, in addition to all the other non-negotiated terms, to become a licensee and stay one, you have to grant a license to others
- What is licensed? For what use?
- That required license likely implies patents owned or controlled, to the extent necessary to use the software (MDLW or entire work?)
- License may not cover all further derivative works made by those using your software for further development

Patent License

- Explicit grants of patent license now made in OS licenses, e.g.:
- Apache License (§3)
- Mozilla Public License (§§2.1, 2.2)
- IBM Public License (§2b)
- Presence of patent license brings license scope and termination issues

Patent License – Apache License

3. Grant of Patent License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted.

Patent License – Apache License

Apache patent license termination (retaliation):

If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

Open Source Patent Termination Implications

- OSS license terminates if licensee makes a patent claim
- Different termination triggers
- What patents affected? What defendants?
- Reflects desire to neutralize patents in OSS area

Other Open Source/Patent Issues

- http://www.patent-commons.org/
- “The Patent Commons Project is dedicated to documenting the boundaries of The Commons – a preserve where developers and users of software can innovate, collaborate, and access patent resources in an environment of enhanced safety, protected by pledges of support made by holders of software patents.”

Patent Commons

- Contributors: HP, Computer Associates, Ericsson, IBM, Nokia, Novell, Red Hat, Sun, OSDL, Jboss, Univ. of Penn., Open Invention Network, Microsoft, Open Logic, Oracle, Blackboard
- Types of patents contributed:

http://www.patent-commons.org/commons/patentsearch.php

Patent Commons

Compression, Encryption, and Access Control	Internet, eCommerce, and Industry Specific
Database and Data Handling	Miscellaneous
Data Processing Programming	Multi-Processing
Human Interfacing	Networking and Network Management
Human Language Processing	Software Development & Object Technology
Image Processing and Video Technology	Storage Management
Interfacing	

Example: IBM's Contribution

- 500 patents
- For what use? Any duties on users?
- IBM's Pledge
- "The pledge will benefit any Open Source Software. Open Source Software is any computer software program whose source code is published and available for inspection and use by anyone, and is made available under a license agreement that permits recipients to copy, modify and distribute the program's source code without payment of fees or royalties. All licenses certified by opensource.org and listed on their website as of 01/11/2005 are Open Source Software licenses for the purpose of this pledge."

IBM's Contribution - Pledge

- "It is our intent that this pledge be legally binding and enforceable by any open source software developer, distributor, or user who uses one or more of the 500 listed U.S. patents and/or the counterparts of these patents issued in other countries."
- "IBM hereby commits not to assert any of the 500 U.S. patents listed below, as well as all counterparts of these patents issued in other countries against the development, use or distribution of Open Source Software."
- Pledge is subject to the exception provided below

IBM's Contribution - Exception

- In order to foster innovation and avoid the possibility that a party will take advantage of this pledge and then assert patents or other intellectual property rights of its own against Open Source Software, thereby limiting the freedom of IBM or any other Open Source Software developer to create innovative software programs, or the freedom of others to distribute and use Open Source Software, the commitment not to assert any of these 500 U.S. patents and all counterparts of these patents issued in other countries is irrevocable except that IBM reserves the right to terminate this patent pledge and commitment only with regard to any party who files a lawsuit asserting patents or other intellectual property rights against Open Source Software

Relevance of Commons?

- Only for open source licensors
- Clarifies the question of what patents are actually licensed – no OSS license lists them
- Some Contributors specify a particular standard or technology supported by the Commitment
- This is, in effect, a field of use license

Open Source Pitfalls

- What happens if GPL applies to a software development in a company? Enforced disclosure remedy; thus “loss” of derivative code? Breach of GPL and loss of license?
- Enforcement actions or lawsuits, actual copyright owners hard to find and position as plaintiffs; often led by Software Freedom Law Center
  - BusyBox litigation against Verizon Communications, Westinghouse, JVC and others
  - FSF v. Cisco (Linksys routers)
- Enforceability of licenses? So far, enforceable.
- M&A, venture capital: buyers, investors and lenders want assurances that assets have not been devalued due to binding effects of or non-compliance with open source licenses

Implementing An Open Source Compliance Policy

- A “just say no” policy is impractical
  - Likely to be perceived as lacking understanding
  - Obstacle in recruiting and retaining developers?
  - Slows down development and increases costs
  - Drives use of open source underground
- Key components of compliance policy
  - Identify stakeholders
  - Education, especially among developers
  - Management of code integrity
  - Procedures for assessing and approving uses

M&A Strategies for Open Source

- For sellers
  - Have policy in place to anticipate OSS inquiries from buyers/lenders
  - Inventory all code and be prepared to show all OSS with applicable licenses, facts of derivation or “work with” relevant to license
  - Show understanding of license terms, including disclosure requirements and compliance with terms
  - Understand how to do a permitted redistribution
  - Be prepared with remedial measures to address deficiencies
  - Push back against open source paranoia, especially in reps and warranties

M&A Strategies for Open Source

- For buyers, due diligence with questions, conversations with coders
  - Code audit by Black Duck, Palamida, Protecode
  - Focus on distributed or planned to be distributed software; prior distributions may also present problems
  - Have techs who understand how software interacts available to understand facts when GPL governs
  - Understand extent of code capture/non-compliance

Questions?

- Many OSS licenses, almost all raise some issues
- http://opensource.org/
  - Lists and gives the text for all licenses it has approved
  - http://opensource.org/faq
    - E.g., Can I stop "evil people" from using my program?
      No. The Open Source Definition specifies that Open Source licenses may not discriminate against persons or groups. Giving everyone freedom means giving evil people freedom, too. Fortunately, there are other laws that constrain the behavior of evil people.
- hemphill.stuart@dorsey.com
- Lancaster.Peter@dorsey.com