DLA PIPER

EVERYTHING MATTERS

Managing Open Source Software 2010: Best Practices

Mark Radcliffe, Partner
DLA Piper, Silicon Valley Office
mark.radcliffe@dlapiper.com
www.lawandlifesiliconvalley.com/blog

DLA Piper

- 3,500 attorneys
- Top global law firm
- Offices in over 25 countries and 65 cities
- Over 25 years in Silicon Valley
- Over 250 corporate attorneys in the US
- Over 400 lawyers in intellectual property in the US (200 in the US)
- From start-ups through Fortune 100
  - Sun Microsystems
  - Zimbra
  - SugarCRM

Market Trends

- Software development has changed forever
  - Internet, community development & open source software (OSS) licensing
  - Componentization and re-use
- Recent survey's confirm OSS has gone mainstream
  - "When it comes to technology investments, OSS adoption now rises to the top"….."Q4 2008 makes it clear that open source software (OSS) is a top priority for software development professionals in 2009."

    Jeff Hammond, Forrester, *Open Source Software Goes Mainstream*, April 2009
- While OSS has gone mainstream, management and policy lags behind
  - A "don't ask, don't tell" pact obscures the reality of OSS use (Forrester)
  - Black Duck Survey at SD West (March 11, 2009):
    - Only 22% of those surveyed reported that their organizations have explicit management policies and procedures in place
    - Only 40% of larger companies with more than 500 developers had explicit management policies

Why Do You Care?

- Litigation: compliance is now an important issue
  - BusyBox suits
- Litigation: remedies are more clearly established after Jacobsen, copyright remedies such as injunctive relief and statutory damages are available
- Your customers care: they are asking for BOM
- Your potential acquirer cares: some companies have separate open source diligence process
- More participation in open source communities: what are your employees contributing?

Key Issues

- Multi-source development using open and closed source software is different than traditional development
- Need a policy for open source use
  - Where and how can open source component can be used
  - Approved licenses and components
    - Can vary by use case
- Cross-functional collaboration
  - More functions are involved
  - Development, Legal, Planning, Management
- Integration with development and standard business processes

Basic Legal Analysis

- If you are acting within the scope of the license
  - You are licensed
  - A license is a defense to a claim of infringement
- If you act outside the scope of the license, or breach the terms of the license so that the license is terminated
  - You are unlicensed
  - You may be an infringer
  - You can be forced to cease activities beyond scope of the license depending on how the license is drafted, see *Jacobsen*
- The Critical Question is:
  - Can I comply with the terms of the license under which the code was made available?

Remedies for Breach of License

- License terms effect which remedies apply
  - Copyright Infringement
    - Injunction available
    - Statutory damages
  - Breach of Contract
    - Monetary damages
    - Injunctions rare
- Sun Microsystems, Inc. v. Microsoft Corp., 188 F.3d 1115 (9th Cir. 1999)
  - "Whether breach of license is actionable as copyright infringement or breach of contract turns on whether the provision breached is a condition of the license or mere covenant."

Critical Decision: Jacobsen

- Model railroad software for hobbyists
  - Developed by professor at Lawrence Livermore Labs
  - Artistic License 1.0 (rarely used and now replaced by Artistic License 2.0)
  - Katzer stripped out copyright and attribution notices
- Complicated claims: patents/copyright/license/unfair competition/domain name
- Remedies Issue for Breach of Copyright Licenses
  - Covenant is breach of contract: monetary damages only
  - Conditional/restriction on license scope (contract & copyright infringement): monetary damages, injunctive relief/statutory damages/attorney's fees
- CAFC Decision
  - Non economic obligations such as notices/attributions can be enforced
  - Wording is critical: "provided that" or "conditional"
  - Applies to both open source and proprietary licenses

German Cases

- Focus of Welte of gpl-violations.org

  Enforcement according to website includes Fortinet, Securepoint, Siemens, Fujitsu, Asus, Belkin Gigabyte Technologies, TomTom, Longshine, ARP Datacom, Edimax (settlements typically secret)
- Welte now allied to FSF Europe
- First jurisdiction to rule on GPLv2 (District Court of Munich in 2002)
- Cases:
  - Distribution of Linux-based devices (e.g. VoIP phone)
    - without providing a copy of GPLv2-license terms
    - without making available the source code
  - Injunctive proceedings

BusyBox Compliance Campaign by SFLC

- Monsoon
- Xterasys
- High Gain Antenna
- Verizon
- SuperMicro Computer
- Bell Products

What is Open Source? Open Source Initiative

- Who is the OSI (Open Source Initiative)?
  - The OSI are the stewards of the Open Source Definition (OSD) and the community-recognized body for reviewing and approving licenses as OSD-conformant.
- OSI lists 72 licenses which OSI has approved as being “Open Source”
- Three types of open source licenses:
  - Reciprocal Licenses: General Public License, Mozilla Public License, Common Public License
  - Notice: Apache License, BSD, MIT
  - Other: NASA Open Source
- Open Source Definition
  - 1. Free 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

General Public License Version 2

- GPLv2 first published in 1991 (final version of GPLv3 published 6/29/2007)
- Key Terms of GPLv2
  - Right of customers to modify and distribute modification under GPL
  - Non-exclusive
  - Obligation to distribute (can charge, but not pass through this obligation)
  - Any “work based on the program” is subject to GPL
  - Must include source code
  - Automatic termination

General Public License Version 2

- Contains political statements
- Scope of “based on” work
  - Derivative work analysis
  - Dynamic vs. static linking
  - Collective work
- Disclaimer of all warranties
- Disclaimer of liability
- Patent license: uncertain (FSF Position: “Implied License”)

BSD

Copyright (c) <YEAR>, <OWNER>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License

- Not viral or “copyleft” like GPLv2
- Separate copyright and patent licenses
- Disclaimer of all warranties
- If an entity provides additional warranties, the entity must indemnify all other contributors
- Must keep notices intact

Lesser General Public License LGPLv2

- "Library software" *is a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables*
- "Works based on library"
  - *Either the Library or any derivative work under copyright law*
  - License: GPLv2 terms
- "Work that uses the Library"
  - *Designed to work with the Library by being compiled or linked with it*
  - *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.*
  - License: Section 6
    - License object code under any terms you wish
    - Provide for reverse engineering
    - Include Library copyright notices
    - GPLv2 obligations relating to source code of library
- "Small Bits" exception to Section 6 (LGPLv2 does not apply)
  - Numerical parameters
  - Data structure layouts and accessors
  - Small macros and small inline functions

What to do?

- Open Source is Ubiquitous
  - Needs to be managed
  - Process is critical
- Cross functional
  - Product Planning/Management
  - Legal, Security & Export Compliance
  - Engineering
- Integrated Processes
  - Component Management
  - License Management
  - Release Management
    - Release Planning
    - Release Delivery

Golden Rule Details

- Systemic
  - Baked in to the culture & workflow
  - Event Driven
    - Component approval request
    - Planning a release
    - Accepting a code drop from a vendor/outsourcer
    - Performing a build
    - Creating a release
- Embrace Supply Chain Techniques
  - ERP systems brought together different users and processes
  - Workflow automates task creation
    - Notifications
    - Process Monitoring
  - Central repositories of data
  - Business Process Integration is the key

Scope of Coverage

- Inbound Licenses
  - Internal use which may be in the form of tools, operating systems and other network infrastructure or applications
- Outsourcing and SAAS
  - Software which you depend upon but never bring into your data center
- Outbound
  - Contributions by the company or contributions by employees, or in the form of products or projects
  - Alone or combined with closed source offerings
- Acquisitions
  - All of the above by a target entity

Open Source Program Elements

1. Published Policy
   - Created via Cross Functional Team
   - Organization is educated on the policy
2. Open Source Process Owner
   - Keeps the wheels running
   - Grant certain types of approvals
3. Approval Processes
   - Component Review & Approval
   - Sensitive to Use: internal/external/products
   - License Review & Approval
   - Release Plan Review & Approval
4. Monitoring & Tracking Process
   - Component Verification
   - Security Notifications
   - Component Upgrade Notifications
   - Application to contractors/outsource vendors
5. Obligation Verification Process
   - Ensure using approved components… and…
   - Meeting the license and business obligations
   - Current reporting for responsive due diligence request

Determine Policies

- Define criteria for approved software
  - Licenses
  - Use (internal/product/website)
  - Sources
  - Support
  - Other
- Define criteria for unapproved software
- Scope of application: internal development, independent contractor, outsource vendors, M&A
- Define conditions for participating in the Open Source Software development
- Employee Education
  - No compliance without education

Sample OSS Policy Contents

Policy component	What it should specify
Goals of OSS adoption	Justification for using OSS (e.g., cost avoidance, speed, performance, quality)
Acquisition processes: • Method of procurement • Distribution policies • Support policies • RACI matrix	• How will you acquire OSS components? • Where are they downloaded from? • How is dependent code made available? • What's the strategy for providing support? • Who is responsible, accountable, consulted, informed?
Rubric for business case	• How will you determine the total cost of ownership? • What performance service-level agreements are needed?
Guidelines for appropriate use including: • License classification • Usage restrictions • Reporting requirements • Derivative use • Remediation policies	• What are the specific guidelines for developers? • What OSS licenses can be used and where? • When should OSS not be used? • How do projects report their use? • How are modifications handled? • What is done when unreported use is detected?

Source: February 2, 2009, "Best Practices: Improve Development Effectiveness Through Strategic Adoption Of Open Source" Forrester report

Forming a Compliance Core Team

- Legal
  - Perform review of identified components
- Open Source Process Owner
  - Appoint a person with overall responsibility
- Business / Product Perspective
  - Prioritize products (by release) for analysis
- Technical / Lead Architect
  - Integrate analysis and review with the development process
  - Identify code based on automated discoveries
- Project Management
  - Coordinate resources
  - Drive the project plan
  - Resolve issues

Common Implementation Issues

- Legalese: make it understandable
- General policy intended for certain products/business model/groups
- 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

ALI Principles of the Law of Software Contracts: Dangerous New Proposal

- ALI has characterized Principles as “expressing the law as it should be, which may not reflect the law as it is.”
- “These Principles seek to clarify and unify the law of software transactions . . . Instead of restating the law, a Principles project accounts for the case law and recommends best practices . . . a Principles project is not the law unless and until a court adopts it.”

Why Should You Care?

- The Principles propose substantial changes to existing case and statutory law
- Represent a decided shift in the balance of interests underlying existing law
- Written for judges, these proposed changes will not go through legislative review
- ALI is obviously a very prestigious and highly regarded institution and the Principles are likely to be used by judges
- UNCITRAL is already considering adopting for other countries

How did we get here?

- ALI's longstanding policies prohibit the free distribution of draft documents and are not designed to invite public comment
- Many individuals and organizations tried to intercede to delay adoption and to encourage ALI to invite input from the industry and other interested parties

Letter on behalf of OSI and the Linux Foundation

http://www.slideshare.net/markradcliffe/osi-and-linux-foundation-letter

Letter on behalf of Microsoft and the Linux Foundation

http://www.slideshare.net/markradcliffe/microsoft-linux-foundation-letter1

Letter on behalf of ITPEC

http://www.slideshare.net/markradcliffe/acc-itpec-letter-and-discussion-points-re-ali-principles-of-the-law-of-software-contracts-5-11-09x

Major Issues

- Non-disclaimable Warranty of no Hidden Material Defects
  - (Section 3.05)
- Implied obligation to Indemnify against Infringement
  - (Section 3.01)
- Standard Form of Transfer
  - (Section 2.02)
- Disclaimer of Warranties
  - (Sections 3.02, 3.06 and 4.01)

Section 3.05 Other Implied Warranties
No Hidden Material Defect

"A transferor that receives money or a right to payment of a monetary obligation in exchange for the software warrants to any party in the normal chain of distribution that the software contains no material hidden defects of which the transferor was aware at the time of the transfer. This warranty may not be excluded. In addition, this warranty does not displace an action for misrepresentation or its remedies."

The Principles explain that "[a] defect exists if the software is not fit for its ordinary purposes" and that "[n]egligence on the part of transferors in failing to discover defects is not covered by the Section and is the subject of products-liability law." The Principles state, "[s]oftware that requires major workaround to achieve contract-promised functionality and causes long periods of downtime or never achieves promised functionality ordinarily would constitute a material defect." The Principles state that this new, non-disclaimable warranty does not replace a separate claim for misrepresentation.

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
- Monitor adoption of ALI Principles and get involved in lobbying ALI (sign the petition to persuade ALI to permit copies of the Principles to be circulated for comment, contact me at my email address if you are interested)