



# Managing Enterprise Licensing Concerns for Open Source Software

Stephen Walli, VP of Open Source Development Strategy

## Where we're going ...

---

### Licensing Basics

- The Absolute Basics (Intellectual Property 101)
- Free and Open Source Licensing Families (GPL, the Academic Licenses, the Mozilla Licenses)
- “Dual” Licensing
- Collective Works
- Thinking about Contributions

### Managing Risks

- License Compliance
- Lawsuits
- Patents
- Indemnifications
- Insurance

## First, the disclaimer

---

- **I am not a lawyer:** I am a business manager and development manager that has worked around free and open source software for 15+ years
- **This presentation is not “legal advice”:** It is the business experience gained from working with free and open source software, proprietary software, and a mix of both in the enterprise, and various software companies including Microsoft
- When in doubt, talk with your lawyers

## If you only remember ONE thing in this talk ...

---

- It's all just software and software licensing:
  - **There is no appreciable difference between free and open source software and any other software when it comes to licensing and legal risk**
  - **Indeed free and open source software is often less risky**
  - **The transparency may at first make things look more complex**

Always question:

“And that would be different from other proprietary or closed software how?”

## The Absolute Basics

---

- Free and open source software (FOSS) licenses depend upon strong intellectual property (IP) law: The idea that FOSS is IP hostile is a **Myth**
- Intellectual Property is a set of legal “tools” that one strategically applies to intellectual assets: Not every asset needs to be considered “intellectual property”
- Types of IP:
  - **Copyright = How you protect the expression of an idea**
  - **Patent = How you publish an idea in a legally protected way so others may not build it**
  - **Trademark = How you protect the way you identify an asset**
  - **Trade Secret = How you legally protect an idea as a secret**
- Companies use a combination of these tools to protect their product in the marketplace
- Software is considered to be protected by copyright law
- In the U.S. one can create a patent for an idea expressed in software

# Basic Families of Free and Open Source Licenses

---

- Historical Academic Licenses (MIT, Berkeley, and Apache)
- Free software and the General Public License Family (GPL)
- The rise of the Corporate Open Source License (Mozilla, IBM)

## Academic Licenses (Berkeley, MIT, and Apache)

- In the early 1980s, the Computer Systems Research Group at UC Berkeley released their UNIX distribution using a license that is referred to as the BSD license:
  - **Enables the user to do ANYTHING with the software**
  - **Does not require any derivative to be licensed the same way or the source to be published for any modifications**
  - **Requires the copyrights to be maintained**
  - **Disclaims any warranties just like proprietary licenses**
- BSD software is at the heart of FreeBSD, NetBSD, OpenBSD, the Windows XP network stack and Mac OS X.
- This license model was later followed by MIT with the pervasive X11 Windowing system and Project Athena
- The Apache Software License was originally modeled this way in the early 1990s since the work came out of the academic community
- Apache 2.0 was a complete rewrite to account for current concerns around contributions and patents, but remained true to the original principals

# Free Software and the General Public License

- Software Freedom (“Free” as in Speech)
  - The freedom to run the program for any purpose
  - The freedom to study how the program works and adapt it to your needs
  - The freedom to redistribute copies so you can help your community
  - The freedom to improve the program, and release your improvements to the public so the whole community benefits
- The GPL was first written in 1985 and sets the terms of software freedom
  - If the user distributes the modified software, it must be done under the GPL -- this is the reciprocity requirement
  - If the user uses any of the software in their own programs and distributes them, they must license the entire work under the GPL -- this is where the concept of a virus is attached to the GPL
  - Disclaims any warranties just like proprietary licenses
- Used in 60%-75% of open source projects including Linux, MySQL

N.B. Nothing forces you to expose your own source: the license contains its own redress and requires you withdraw the distribution if you can not comply

## Corporate Open Source Licenses (Mozilla, IBM)

---

- Late 1990s Netscape releases its browser as Mozilla and creates the Mozilla Public License:
  - **Requires derivatives that are the original work plus contributions to be licensed using the MPL, so preserves the reciprocity of the GPL**
  - **Enables the Work to be combined into a Larger Work and re-licensed, so behaves like the academic licenses in allowing “closed” work to happen**
  - **Discusses patent rights**
  - **Disclaims any warranties just like proprietary licenses**
- This is the first “corporate” open source license and reads more like a corporate license
- IBM models the IBM Public License on the MPL which evolved to the Common Public License and then the Eclipse Public License

## “Dual” Licenses

---

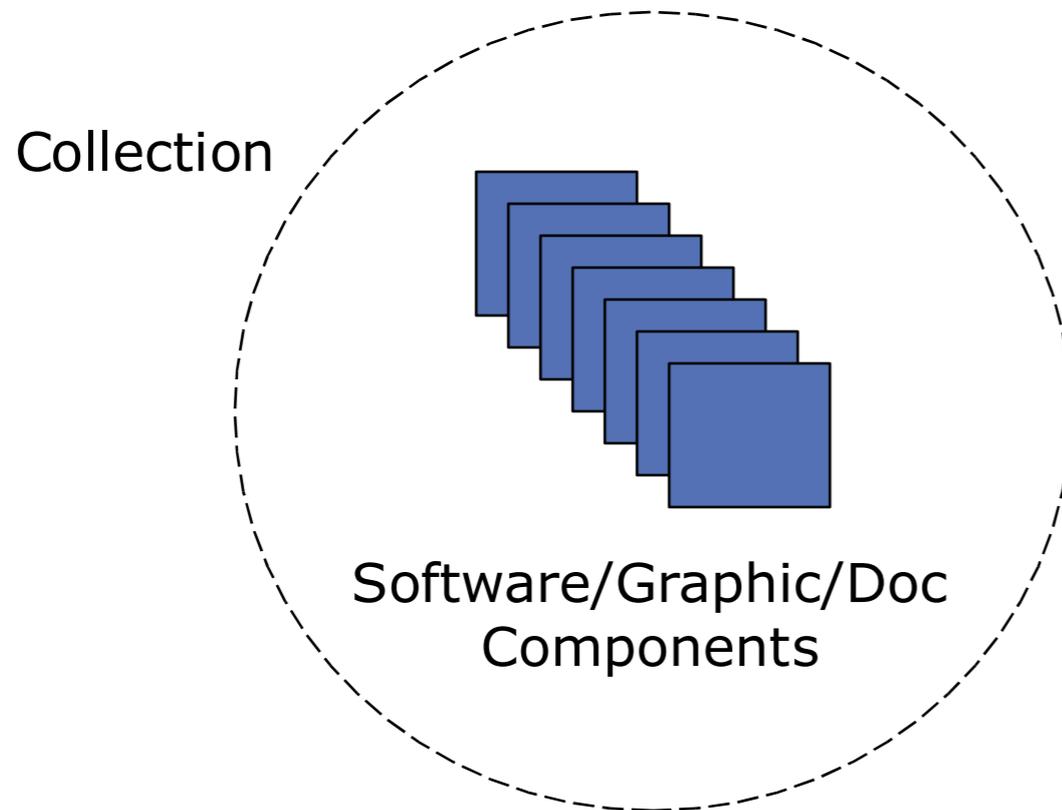
- It is an attribute of IP law that the property holder can license their property as many times and in as many ways to as many people as they choose
- MySQL AB licenses MySQL (the database engine) under both the GPL if the licensee is using or developing free software, or under a MySQL OEM license if the licensee wishes different terms
- This is often referred to as “Dual Licensing” ...
- ... but when you buy a copy of Windows XP in the office supply store it is covered by a Microsoft End User License Agreement (EULA) which is very different from the Enterprise Agreement signed by your corporation for your work machine ...
- ... so “Dual Licensing” isn’t really an free and open source software issue

## Collected Works

---

- One can also copyright a “collection” of works where each work has its own copyright (e.g. an encyclopedia)
- Software products and distributions that contain free and open source components often carry their own licenses:
  - **Red Hat Advanced Server has a license**
  - **Fedora Linux (also produced by Red Hat) has a different license**
  - **Novell OpenSuSE Linux has a different license again**
  - **Microsoft Windows XP is covered in the same way**

# Bringing it all together with an example



## Components:

- Every one is protected by copyright
- Each one may be protected by trade secret
- Patents may be applied to aspects or combinations of components
- Each one might have a different 3rd party license

## Collection:

- Protected by collected work copyright
- Protected by trademark
- A license governs the use of the product (collected work)

## The Free/Open Source Difference:

- The component licenses are generally free and open source licenses (i.e. most source code is accessible and usable)
- There is probably little trade secret protection (i.e. most source code is visible)
- There are probably no software patents

# Community and Contributions

---

- **Community Roles:**
  - Users: They use the software
  - Testers: File bugs in the bug database
  - Contributors: Contribute bug fixes, enhancements, documentation, answer questions, etc.
  - Committers: Developers with check-in access to the software repository
  - Leadership: The developers that keep it all working
- People can have more than one role at any one time
- Contributors generally need to assign or license their contributions to the project in large well run projects.
  - Originality
  - Right to assign
  - To the best of the creators knowledge, nothing infringes others rights
- The project license is an outbound document that says how the software can be used
- The assignment document is an inbound document to provide clear provenance for the software from a legal perspective

## License Management and Compliance

---

- License management and Compliance: The concerns raised over tainting your code, having to publish your own secrets, and infringing property are mostly vendor-centric propaganda.
- The code taint problem isn't directly related to open source, but to software development in general.
- **Open source isn't the problem here:** consider all the other sources of "taint" a company encounters include vendor portals, third party code, and products.
- Publication risk is only triggered on distribution if the GPL is involved and even then, you can withdraw the distribution and correct the code usage.
- Software product companies have different risk profiles compared to enterprises with respect to infringement in general
- Education for Developers, Managers, and Lawyers is key
- Compliance checking tools (i.) can easily miss other taint sources, and (ii.) lead to a false sense of security: Compare the risk and investment in context to other software development process and tool needs

## Getting Sued over Open Source Software

---

- The SCO Group is the data point around which all discussion revolves.
- The Canopy Group (SCO Group parent company at the start of the suit) has a historical business model based on technology acquisition and litigation. They were successful against Microsoft and CA.
- SCO Group began litigation against Autozone and Daimler-Chrysler as a pressure tactic against the IBM suit. These suits were successfully put behind the outcome of the main SCO vs. IBM suit.
- Other vendors with threatened business models have used this as a rallying point for discussions around indemnification and dressed it up in the rhetoric of property ownership.
- Still other vendors have used this as a rallying point to try to sell you tools, insurance, etc.

# The Legal Risks of Proprietary Software

- Companies stand a greater risk of being sued over license counting issues from their commercial software vendors.
- Some proprietary software vendors use the threat of BSA auditing and litigation as a stick.
- Open source licensing **reduces risk** in these areas.



VS.



## Patent Litigation in Reality

- Patents are negotiation tools between vendors
- IP is a supply side consideration; It's not a customer side concern
- A couple of thought experiments:



#1 Did you buy the Honda over the Toyota because it had more patents?

#2 If Toyota then tried to license their patents directly to you: would you?

# Indemnifications and Open Source

- It all started with the SCO Group suit ....

**Offering Indemnifications**

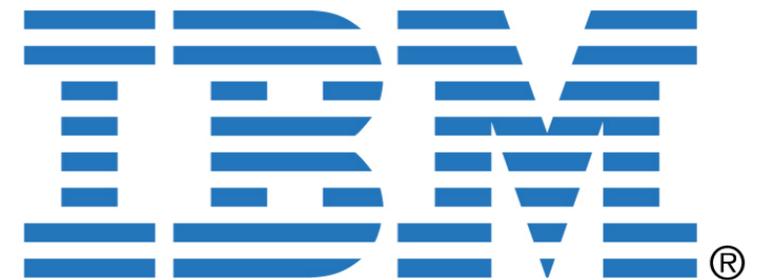


**N**



**VS.**

**Insists Indemnifications  
Unnecessary**



# Insurance

---

- Insure assets (not liabilities) based on real value and measurable risk
  - **One may increase life insurance over time as earning power increases**
  - **One may decrease the auto insurance over time as the value depreciates**
- A Lloyds of London underwriter in conjunction with Open Source Risk Management has recently created an insurance offering for software development companies based on license compliance and the GPL
- Pitched as business interruption insurance in the event that you need to expose your source code or withdraw your product due to GPL taint, and involves compliance inspections by OSRM staff (surprise!)
- One needs to evaluate such insurance against the industry loss data and the opportunity costs of other process improvement or risk management costs

*“All complex ecosystems have parasites.”*

*Cory Doctorow*

## In Summary ...

---

- There is nothing special per se about “free” and “open source” and software licensing
- There is a collection of software licenses that support collaborative development projects and if you are using the source code you need to read the licenses to determine if you’re willing to work with them
- The SCO Group law suit is not a data point
- White papers available at <http://www.optaros.com>
- <http://stephesblog.blogs.com> and [stephe@optaros.com](mailto:stephe@optaros.com)

## Questions?

**This deck is published under a Creative Commons Attribution 2.5 license. The text of the license can be found at <http://creativecommons.org/licenses/by/2.5/>**