

# Intellectual Property Law & Open Source Software

Cliff Schmidt

[cliff@symbiosstrategy.com](mailto:cliff@symbiosstrategy.com)

29 June 2006

Copyright 2006 Cliff Schmidt  
Licensed under [Creative Commons Attribution-NonCommercial-NoDerivs 2.5](https://creativecommons.org/licenses/by-nc-nd/2.5/)

ApacheCon  
Europe 06

# Agenda

- Fundamentals of Intellectual Property Law  
coffee break
- Overview of Open Source Licenses  
keynote  
lunch
- License Spotlight: Apache, GPL/LGPL,  
EPL, and CeCILL  
coffee break
- Beyond the Fundamentals
- IP Best Practices of Open Source Projects

# Disclaimer

- In an effort to convey a lot of detail-rich information in a small amount of time and space...

*these slides abbreviate, edit, and omit many details of intellectual property law and specifics of particular licenses and licensing practices.*

Do NOT use these slide as a substitute for having a lawyer read the actual legal text!

# Who are You?

- Committers at Apache?
- Committers on other open source projects?
- Project Management Committee members?
  
- Managers of committers?
- Lawyers for committers?
- Users of Open Source Software?

# Who am I?

- V.P. of Legal Affairs for the Apache Software Foundation
- Consultant to/participant in:
  - Free Software Foundation's GPLv3 discussion committees
  - Eclipse Foundation's legal policies
  - ObjectWeb Consortium's License & Business Models
  - Open Source Initiative's License Proliferation committee
  - Legal staffs, product/IT groups, execs of companies using OSS
- Not a lawyer
- Also:
  - member of Apache Incubator & Eclipse Technology PMCs
  - member of Apache Board of Directors



# Fundamentals of Intellectual Property Law

Copyright 2006 Cliff Schmidt  
Licensed under [Creative Commons Attribution-NonCommercial-NoDerivs 2.5](https://creativecommons.org/licenses/by-nc-nd/2.5/)

ApacheCon  
Europe 06



# Copyright

- Covered actions
  - reproduce
  - **prepare derivative works**
  - distribute
  - publicly display
  - publicly perform
- Key requirements
  - original expression with some *minimal* creativity
  - fixed in a tangible medium
- Term:
  - 75 years after death of author, or
  - 95 years after publication for corporations/anonymous

# Copyright Registration

- Registration is required before application for infringement
  - to recover statutory damages and attorney fees
  - may be registered after, but within three months, of publication
- Application
  - deposit of work (1st 25pp of source code, + last 25 pp of code)
  - application form
  - \$30 registration fee

# Copyright and License Notices

- Copyright notice no longer required since March 1, 1989
- However, notice defeats an innocent infringement defense, which may be used to mitigate actual/statutory damages.
- Form of notice:
  - claim of copyright (“Copyright” or “Copr.” or “©”)
  - year of first publication
  - name or abbreviation of copyright owner
- Placement of copyright notice for software
  - “user terminal at sign-on”
  - continuously displayed
  - affixed to the box

# Derivative Works

**“a work based on one or more preexisting works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgement, condensation, or any other form in which a work may be recast, or adapted. A work consisting of editorial revisions, annotations, elaborations, or other modifications, which, as a whole, represents an original work of authorship, is a ‘derivative work’”**

Section 101 of U.S. Copyright Act

# Copyright

- Not everything is protected by copyright law
  - idea-expression merger
  - scènes à faire
  - de minimus work
- Indications of copyright infringement
  - substantial similarity, AND
  - access to infringed work
- Bottom line:
  - must have a license from the author/owner to take any of the covered actions (beyond fair use)

# Collective Works

- Compilation is:
  - work formed by the collection and assembling of preexisting materials or of data that are selected, coordinated, or arranged in such a way that the resulting work as a whole constitutes an original work of authorship. Compilations include collective works.
- Collective Work is:
  - periodical issue, anthology, or encyclopedia in which a number of contributions, constituting separate and independent works in themselves, are assembled into a collective whole.
- Choices in selection, coordination, and arrangement must involve some originality (e.g. not chronological order)

# Collective Works

- Author owns individual work, but copyright in collection vests in the author of the collection.
- Author of collective work must acquire rights of reproduction and distribution from other authors for use in collective work of later version.

# Patents

- Covered actions
  - make
  - use
  - sell
  - have made
  - import
  - offer to sell
- Key Requirements
  - novel
  - non-obvious
  - has utility
  - described in detail
- Requires patent application and approval by patent authority (e.g. EPO)

# Patents

- Described invention may apply to multiple implementations
  - (patents are not written to protect a particular piece of software)
- Limited Monopoly
  - provides the right to exclude others from the above actions for a limited time
  - covers processes, designs, machines, article of manufacture (differs across international laws)

# Patents

- Implied Patent License
  - Each distributee of a patented article gets a license from the distributor to practice any patent claims held by the distributor that cover the distributed article.
- Bottom Line
  - independently created inventions still require a license from any valid, pre-existing, applicable patent
  - very difficult to check novelty of software patents
  - patent law does not discriminate between open source and proprietary software
    - easier to see infringement vs. greater archive of “prior art”

# Impact of Patents

- Cost of Patents
  - ~\$10-20k/patent
  - ~\$3M to litigate a patent law suit
- Owners build defensive portfolios
- Settle with cross-licensing agreements
- Notification requirements:
  - Many jurisdictions in Europe require patent holder to provide notification of infringement and an opportunity to remedy prior to suing for damages

# Patents in Europe

- Still must be novel and “level of inventiveness” (non-obvious) with utility, described in detail
- Same 20-year term
- Many patent offices / issuing authorities:
  - European Patent Office
  - National offices
  - WIPO offices for forward
- What’s this about patents on computer programs not being allowed *as such*?

# Nationally-Issued Patents in Europe

- each nation has a patent office that may issue patents for their country's jurisdiction and no other
- each nation's courts determine the validity of such patents
- patents are not honored in other EU nations

# Patent Agreements in Europe

- Munich Convention on European Patents of 1973
  - also known as “European Patent Convention (EPC)”
  - not an EU measure
    - but nearly all EU members have signed (not Malta)
  - created European Patent Office (EPO)
  - EPO reviews applications for a “European Patent”
    - opposition procedure allows anyone to challenge during grant stage
    - act as “bundles of national patents”
    - may be found to be invalid in one country and upheld by another

# Patent Agreements in Europe

- Patent Cooperation Treaty (PCT) of 1978
  - signed by all EU member nations, except Malta
  - also signed by U.S.
  - governed by WIPO
    - send one application to Geneva
    - check which territories you wish to apply to
    - WIPO office does administrative checks and forwards application to each patent office
      - EPO is option, in addition to national patent offices of all signatory nations

# European Patents and Software

- EPC states patents on computer programs “as such” are excluded
  - EPO interprets this to mean only technical software is patentable (broad and not well defined)
  - EPO has issued several thousand software patents
- Since 1978, EPO is estimated to have issued ~30,000 software patents

# European Patents and Software

- Directive on the Patentability of Computer-Implemented Inventions
  - proposed in Feb 2002, but rejected overwhelmingly by European Parliament in 2005
  - still national courts, but can seek ruling from European Court of Justice
  - debate: would include all software? only software in hardware?

# European Patents and Software

- European Community Patent
  - saving for all EU territories from \$60,000 to ~\$25,000
  - cost savings partly due to translation from 20 to 3 languages
  - either be valid or fail across all member states

# European Patent Litigation Agreement (EPLA)

- creates
  - a "European Patent Court" and
  - a "European Patent Court of Appeal"
- attempting to address same issues of separate national courts

# Trademarks

- Identifies the origin of a product or service (service mark)
  - includes brand names, symbols, shapes, numbers, slogans, sounds, and smells
  - identify and distinguish one company's products from another
- Protects both
  - trademark owner (erosion of brand value), and
  - public (deception of source and quality)
- Allows preventing others for using confusingly similar marks

# Trademarks

- Registration is not required (protection for “first use”), but helps
  - other countries give more weight to registration
- *Few open source licenses grant any trademarks rights.*

# Community Trade Mark (1996)

- regulated by series of EC regulations
- 10 years at a time, but in perpetuity
- rejected on absolute grounds not relative grounds (not worried about similarity with others)
- Madrid Protocol of 1989 - forwards from WIPO to national offices

# Trade Secrets

- Any formula, pattern, device, or compilation of information that derives economic value by being secret
- Techniques that can be reverse engineered (by studying product or documentation) cannot be kept secret.
- Based on how many people know/precautions taken to guard it

# Trade Secrets

- Mutually exclusive with patents
  - secret could be patentable or not
- No time limit
- Protected by state law (varies from state to state)

# Comparison to IP Laws of other Countries

- Moral Rights
  - foundation of copyright law in many EU nations
  - moral rather than economic rights
  - work is invested with artist's spirit/personality
  - inalienable and unwaivable rights include:
    - integrity of the work
    - attribution of the artist
    - disclosure to the public
- U.S. Visual Artists Rights Act of 1990
  - addresses Berne Convention's attention to moral rights -- not for software

# Applying IP Law to Open Source Software

Copyright 2006 Cliff Schmidt  
Licensed under [Creative Commons Attribution-NonCommercial-NoDerivs 2.5](https://creativecommons.org/licenses/by-nc-nd/2.5/)

ApacheCon  
Europe 06



# Why Care about Other Licenses?

- If your open source project is required to release code under a single license...why should you care about the others?
  - How will your software work with other open source software?
  - Will you ever include other open source software in your distributions?
  - Will your project ever revise or replace the current license?

# The Open Source Definition

## 1. Free Redistribution

- including as component of other software

## 2. Source Code

- included or available, in preferred form

## 3. Derived Works

- modifications can be licensed the same

# The Open Source Definition

4. Integrity of Author's Source Code
  - may distribute derivative works, but might need name change or use patch files
5. No Discrimination Against Persons or Groups
  - nor can you give one company extra rights
6. No Discrimination Against Fields of Endeavor
  - cannot forbid use in particular industries

# The Open Source Definition

## 7. Distribution of License

- all 3rd parties; no other required license

## 8. License Must Not Be Specific to a Product

- same rights if extracted and used elsewhere

## 9. License Must Not Restrict Other Software

- if simply distributed on same medium

## 10. License Must Be Technology-Neutral

- cannot require “I accept” button

# How to Distinguish 58 OSI-Approved Licenses?

- Academic Free License (AFL)
- Adaptive Public License
- Apache License V2.0
- Apache Software License
- Apple Public Source License
- Artistic License
- Attribution Assurance License
- BSD License
- Common Development and Distribution License
- Common Public License
- Computer Associates Trusted Open Source License
- CUA Office Public License Version 1.0
- Eclipse Public License
- Educational Community License
- Eiffel Forum License
- Eiffel Forum License V2.0
- Entessa Public License
- EU DataGrid Software License
- Fair License
- Framework Open License
- GNU General Public License (GPL)
- GNU Library/Lesser General Public License (LGPL)
- Historical Permission Notice and Disclaimer
- IBM Public License
- Intel Open Source License
- Jabber Open Source License
- Lucent Public License (Plan9)
- Lucent Public License Version 1.02
- MIT License
- MITRE Collaborative Virtual Workspace License
- Motosoto License
- Mozilla Public License 1.0 (MPL)
- Mozilla Public License 1.1 (MPL 1.1)
- NASA Open Source Agreement
- Naumen Public License
- Nethack General Public License
- Nokia Open Source License
- OCLC Research Public License 2.0
- Open Group Test Suite License
- Open Software License
- PHP License
- Python License (CNRI Python License)
- Python Software Foundation License
- Qt Public License (QPL)
- RealNetworks Public Source License V1.0
- Reciprocal Public License
- Ricoh Source Code Public License
- Sleepycat License
- Sun Industry Standards Source License (SISSL)
- Sun Public License
- Sybase Open Watcom Public License
- University of Illinois/NCSA Open Source License
- Vovida Software License 1.0
- W3C License
- wxWindows Library License
- X.Net License
- zlib/libpng License
- Zope Public License

# One Dozen Significant Open Source Licenses

<b>License</b>	<b>SF.net projects</b>	<b>first approved</b>
1 GNU General Public License (GPL)	48827	Feb-89
2 GNU Library or Lesser General Public License (LGPL)	8359	Jun-91
3 BSD License	5306	Jun-89
4 Apache License V2.0 + ASL	1825	Feb-95
5 MIT License	1383	Sep-87
6 Artistic License	1288	Mar-91
7 Mozilla Public License 1.1 (MPL 1.1) + 1.0	1238	Jan-98
8 Common Public License	635	Aug-02
9 Open Software License	439	Oct-02
10 Academic Free License (AFL)	410	Oct-02
11 <del>zlib/libpng License</del>	<del>383</del>	<del>May-95</del>
12 <del>Qt Public License (QPL)</del>	<del>232</del>	<del>Nov-98</del>
13 Eclipse Public License	155	May-04
14 Common Development and Distribution License	69	Jan-05
15 <del>University of Illinois/NCSA Open Source License</del>	<del>44</del>	<del>Oct-87</del>

March 2006

ApacheCon  
Europe 06

Copyright 2006 Cliff Schmidt

Licensed under [Creative Commons Attribution-NonCommercial-NoDerivs 2.5](http://creativecommons.org/licenses/by-nc-nd/2.5/)

# Distinguishing Features

- Degree of reciprocity
- Sublicensing options
- Patent Grant
- Patent Retaliation
- Must provide complete source or ability to get it
  - like MPL, CDDL, CPL, EPL, OSL, etc
- Prohibition on charging royalties on original work
  - GPL, LGPL, Artistic License
- Prohibition on charging royalties on derivative works
  - GPL, LGPL

# Distinguishing Features

- Requirement to prominently note modifications to each version
  - Yes: GPL, LGPL, Apache, MPL, CDDL, Perl, AFL, OSL
  - No: CPL, BSD, MIT, EPL
- Can impose new license restrictions on distributed/modified copies
  - Yes: MPL, CDDL, CPL, EPL, Apache, Perl, MIT, BSD
  - No: GPL, LGPL, AFL, OSL

# Definitions

- License
  - permission to commit some act that would otherwise be unlawful
- Sublicense
  - grant of a portion or all of the rights granted to the original licensee
- Reciprocity
  - requirement to license distribution of derivative works under the same license as the original work
- Patent Retaliation
  - termination of copyright or patent grants as a result of patent action

# Defining Categories

- Reciprocity Reach
- Sublicensing Restrictions  
and ability to charge royalties
- Patent Grant
- Patent Retaliation

# Reach of Reciprocity

0: none

1.0: file reciprocity

1.5 module *restrictions*

2.0: module reciprocity

-or- (file reciprocity + hosting impact)

3.0: derivative reciprocity

4.0: (derivative reciprocity + container *restrictions*)

-or- (derivative reciprocity + hosting impact)

5.0: container reciprocity

# Sublicensing Options

- 0: explicitly rights to sublicense source/binary under other agreements
- 1.0: implicit ability to sublicense source/binary under other agreements
- 2.0: {empty}
- 3.0: sublicense source:same license; binary: other agreements okay
- 4.0: sublicense source/binary under same terms; no royalties allowed
- 5.0: no sublicense - direct license from author; no royalties allowed

# Patent Grant

0: explicit patent grant

1.0: implicit, not well-defined

2.0: no patent grant

# Patent Retaliation

- The two key aspects of a patent retaliation clause are:
  - 1) what triggers the termination of rights, and
  - 2) which rights are terminated.
- The trigger tends to be the institution of patent litigation, but the question is:
  - against whom and for what?

# Patent Retaliation

## 1. Patent Litigation Trigger:

- a. against anyone for the work  
(or a Contribution incorporated within the Work\*)
- b. against anyone for anything based on the covered work
- c. against the contributor for the work
- d. imposition of conditions contradicting license
- e. against the contributor for any software  
(or any hardware or device\*)
- f. against the contributor for anything,  
if the contributor didn't sue first

# Patent Retaliation

## 2. Results in Termination of:

- v. patent grants from the contributor being sued
- w. copyright & patent rights from contributor being sued
- x. patent grants from any contributor
- y. the right to distribute the work
- z. the entire license  
(or the just the rights to 'modify and run the Program'\*)

# Patent Retaliation

- Triggers and Results
  - AFL,OSLv2.1+: a -> z
  - Artisticv2beta15: a -> z
  - Apachev2: a\*-> x
  - EPL,CATOSL: a -> x
  - CPL,IPL,Lucent: a -> x + e -> v
  - MPL,SPL,Nokia,Richoh: c -> w + e\*-> v
  - Reciprocal: c -> w + a -> z
  - Motosoto: c -> w
  - CDDL: c -> z
  - Apple,Sybase,Real: f -> z
  - GPLv2,LGPLv2.1,OCLC: d -> y
  - GPLv3dd1: d -> y + b -> z\*

# Patent Retaliation

- GPLv3 Section 7e says Terminations v-z are okay if under stated conditions that are a subset of the following trigger cases:
  - 1. f
  - 2. a,b,c,d
- Licenses that appear to conflict with Section 7(e):
  - Adaptive Public License
  - Common Public License
  - IBM Public License
  - Lucent Public License (Plan9 and v1.02)
  - Mozilla Public License 1.1
  - Nokia Open Source License
  - Ricoh Source Code Public License
  - Sun Public License

# Patent Retaliation: Litigation Trigger

- 0: None
- 1.0: against contributors for work
- 2.0: against anyone for work
- 3.0: any restrictions on royalty-free distribution
- 4.0: against contributor for any software
- 5.0: against contributor for any software, hardware, or device

# Patent Retaliation: Termination Details

0: None

1.0: Loss of patent grants from contributor

2.0: Loss of patent grants from all contributors

3.0: {empty}

4.0: May not distribute

5.0: Loss of all copyright and patent rights

# Note about Patent Retaliation Calculations

- Patent Retaliation score = average of:

*litigation trigger score*  
&  
*termination details score*

- When two different retaliations in same license  
total score =  $\log_2 (2^{\text{score1}} + 2^{\text{score2}})$
- However, the following may warrant a score of 0, rather than 2:
  - removing rights for claims against the work by any user of the work

# Quick Look at a Dozen Open Source Licenses

- GPL
- LGPL
- Artistic License v1
- Artistic License v2
- CPL
- EPL
- Apache License
- MPL
- CDDL
- BSD
- MIT
- AFL
- OSL

# Quick Look: GPL v2

- Reach: 5
  - reciprocity covers container/derivative works
- Sublicense: 5
  - no sublicense; license granted from original author upon each distribution
- Patent Grant: 1
  - no explicit patent grant, but license implies one
- Retaliation: 3.5
  - no distribution unless everyone can distribute RF

# Quick Look: LGPL v2.1

- Reach: 4
  - covers containers/derivative works, but only restricts (does not require reciprocity) of licenses
- Sublicense: 5
  - same as GPL
- Patent Grant: 1
  - same as GPL
- Retaliation: 3.5
  - same as GPL

# Quick Look: Artistic License v1

- Reach: 1.5
  - “provided that you do at least ONE of the following:  
{make “Freely Available”} or {rename non-standard executables}
- Sublicense: 1
  - “may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution”
- Patent Grant: 2
  - none
- Retaliation: 0
  - none

# Quick Look: CPL v1.0

- Reach: 2.5
  - “do not include additions to the Program which:  
(i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and  
(ii) are not derivative works of the Program.”
- Sublicense: 3
  - may sublicense object code under other agreements if license includes notices and disclaimers

# Quick Look: CPL v1.0

- Patent Grant: **0**
  - explicit grant for contribution and combination
- Retaliation: **3.3**
  - patent litigation against a Contributor with respect to a patent applicable to software:
    - then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate
  - patent litigation against any entity alleging that the Program itself infringes such Recipient's patent(s):
    - then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

# Quick Look: EPL v1.0

- Reach: 2.5
  - same as CPL
- Sublicense: 3
  - same as CPL
- Patent Grant: 0
  - same as CPL
- Retaliation: 2
  - only second part of patent retaliation clause
  - very similar to Apache License

# Quick Look: Apache License v2

- Reach: 0
  - not reciprocal
- Sublicense: 0
  - “may provide additional or different license terms and conditions of Your modifications”
- Patent Grant: 0
  - explicit grant for contribution and combination
- Retaliation: 2
  - very similar to EPL

# Quick Look: MPL v1.1

- Reach: 1
  - file-based reciprocity
- Sublicense: 3
  - redistribute executable under own terms
- Patent Grant: 0
  - explicit patent grant for “contributor version”
- Retaliation: 4
  - lose Contributor’s patent & *copyright grants* if assert claim against Contributor for work
  - lose Contributor’s patent grants if assert patent claim against any “software, hardware, or device”

# Quick Look: CDDL v1.0

- Reach: 2
  - same as MPL 1.1 + “or otherwise make available”
- Sublicense: 3
  - same as MPL 1.1
- Patent Grant: 0
  - same as MPL 1.1
- Retaliation: 3
  - lose *all* patent & copyright grants if assert claim against any contributor for the work

# Quick Look: BSD (New)

- Reach: 0
  - not reciprocal
- Sublicense: 1
  - no mention of sublicense, but often assumed
- Patent Grant: 1.5
  - no patent grant, just the word “use”
- Retaliation: 0
  - none

# Quick Look: MIT

- Reach: 0
  - same as BSD (not reciprocal)
- Sublicense: 1
  - “sublicense” is explicit, sublicense flexibility is implicit
- Patent Grant: 1
  - not explicit terms, but use of words “use” and “sell”
- Retaliation: 0
  - same as BSD (not reciprocal)

# Quick Look: AFL v2.1/3.0

- Reach: 0
  - not reciprocal
- Sublicense: 0
  - “to distribute...under any license of your choice that does not contradict the terms and conditions”
- Patent Grant: 0
  - explicit grant for the work
- Retaliation: 3.5
  - termination of entire license if sue any licensee

# Quick Look: OSL v2.1/3.0

- Reach: 4
  - reciprocal: distribution = *“use...of the Original Work or Derivative Works in any way such that [it] may be used by anyone other than You”*
- Sublicense: 4
  - may only sublicense under same terms
- Patent Grant: 0
  - same as AFL
- Retaliation: 3.5
  - same as AFL

# The Big License Bar Graph

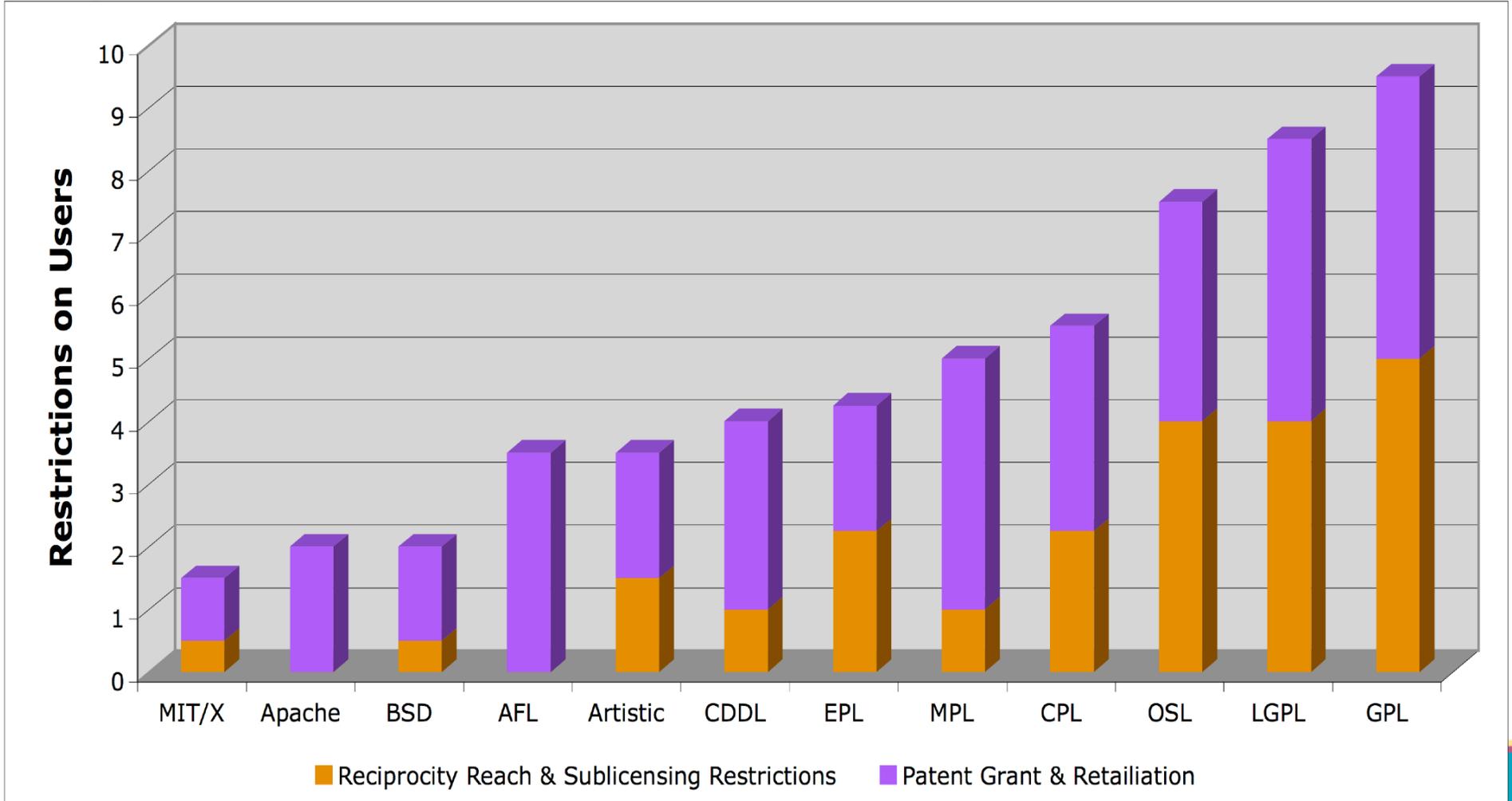
Copyright 2006 Cliff Schmidt  
Licensed under [Creative Commons Attribution-NonCommercial-NoDerivs 2.5](https://creativecommons.org/licenses/by-nc-nd/2.5/)

ApacheCon  
Europe 06



# License Chart Calculations: Simplified

- Reciprocity Reach & Sublicensing Restrictions
  - 0: No reciprocity & explicitly flexible sublicensing
  - 0.5: No reciprocity & flexible sublicensing
  - 1.0: File reciprocity & source sublicensing restrictions
  - 1.5: Module restrictions & minor src/bin sublicensing restrictions
  - 2.0: Module reciprocity & source sublicensing restrictions
  - 3.0: Derivative reciprocity & src/bin sublicensing restrictions
  - 4.0: Derivative/hosting reciprocity & s/b sublicensing restrictions
  - 4.0: Container restrictions & no sublicensing/royalties permitted
  - 5.0: Container reciprocity & no sublicensing/royalties permitted
- Patent Grant & Retaliation
  - Sum grant & retaliation scores (still  $\geq 0$  &  $\leq 5$ )





lunch



# In-depth: Apache, EPL, GPL/LGPL, and CeCILL

Copyright 2006 Cliff Schmidt  
Licensed under [Creative Commons Attribution-NonCommercial-NoDerivs 2.5](https://creativecommons.org/licenses/by-nc-nd/2.5/)

ApacheCon  
Europe 06



# Apache License v2: Reminder

- Reach:
  - not reciprocal
- Sublicense:
  - “may provide additional or different license terms and conditions of Your modifications”
- Patent Grant:
  - explicit grant for contribution and combination
- Retaliation:
  - very similar to EPL

# Apache License v2.0:

## 1. Definitions

- Read/refer to them
  - e.g “Derivative Works”, “Contributor”
- Definitions
  - *Contribution* = “...any work of authorship, including...additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner...”
  - *Submitted* = “...any form of electronic, verbal, or written communication sent to the Licensor...including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems...but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as “Not a Contribution.”

# Apache License v2.0:

## 2. Grant of Copyright License

- Perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable license to all copyright rights
- May sub-license
- Applies to both Source and Object form

# Apache License v2.0:

## 3. Grant of Patent License

- Perpetual, worldwide, non-exclusive, royalty-free, irrevocable license to all patent rights
- May transfer the Work
- “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 termination upon instituting patent litigation over Work against any entity.

# Apache License v2.0:

## 4. Redistribution

- Applies to:
  - source/object, with/without modification
- Conditions:
  - give copy of Apache License
  - prominent notice of change for each modified file
  - source must retain all IP notices
  - include all relevant attribution NOTICES
- Handling original additions/modifications:
  - may add your copyright notice to your modifications
  - offer under other licenses

# Apache License v2.0:

## 5. Submission of Contributions

- Contribution is licensed under Apache License
  - ..unless separate agreement executed

# Apache License v2.0:

## 6. Trademarks

- No license to trademarks other than to:
  - describing origin of work
  - reproducing content in NOTICE file

# Apache License v2.0:

## 7. Disclaimer of Warranty

- Unless required by law, Work on a “AS IS” basis, including:
  - TITLE,
  - NON-INFRINGEMENT,
  - MERCHANTABILITY,
  - FITNESS FOR A PARTICULAR PURPOSE

# Apache License v2.0:

## 8. Limitation of Liability

- Unless required by law, Contributors are not liable for damages arising as a result of using or not being able to use the Work

# Apache License v2.0: NOTICE and LICENSE files

- NOTICE file
  - notifies user of required attributions
  - lists the ASF copyright notice for the collective work
  - does not add additional terms to the license
  - primarily used for “About” boxes
- LICENSE file
  - copy of Apache License v2.0
  - include copy of or link to additional licenses for included non-Apache software within distribution

# Contributor License Agreement

- It's a license - not copyright assignment!
  - “You reserve all right...in and to your Contributions.”
  - applies to all present and future contributions
- Grants copyright and patent license to Foundation and recipients
  - same rights granted to users in the Apache License, including the patent termination clause
- Legally entitled to grant license (see cCLA)
  - submission must disclose any known restrictions (e.g. patent/trademark licenses) - update if turns inaccurate
- No warranty being made

# Corporate Contributor License Agreement

- Allows employer to license directly with ASF,
  - rather than have developers grant rights on behalf of their corporation
- Authorizes which employees are approved to contribute employer's IP
- Employee-committers still sign individual CLAs
  - otherwise, how would Apache have license to a contribution that was actually the employee's own IP?

# Apache License v2.0: NOTICE and LICENSE files

- NOTICE file
  - notifies user of required attributions
  - does not add additional terms to the license
  - primarily used for “About” boxes and start-up messages
- LICENSE file
  - copy of Apache License v2.0
  - copy of any additional licenses for included non-Apache software or link to additional licenses within distribution

# EPL v1.0: Reminder

- Reach:
  - “do not include additions to the Program which:  
(i) are **separate modules** of software distributed in conjunction with the Program under their own license agreement, and  
(ii) are **not derivative works** of the Program.”
- Sublicense:
  - may sublicense object code under other agreements if include notices and disclaimers

# EPL v1.0 Reminder

- Patent Grant:
  - explicit grant for contribution and combination
- Retaliation:
  - patent litigation against any entity alleging that the Program itself infringes such Recipient's patent(s) then
  - such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

# EPL: 1. Definitions

- “Contribution”:
  - initial code and documentation (“the Program”)
  - changes/additions to Program (also becoming “the Program”)
  - does not include additions to the Program which:
    - (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and
    - (ii) are not derivative works of the Program.
- “Contributor”
  - any person or entity that distributes the Program.
- “Licensed Patents”
  - patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with Program

# EPL: 2. Grant of Rights

- Worldwide, non-exclusive, royalty-free, license to
  - all copyright rights + right to sublicense
  - all patent rights + right to “otherwise transfer” all Licensed Patents

“This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution.”

- applies to both source and object form

# EPL: 2. Grant of Rights

- Worldwide, non-exclusive, royalty-free, license to
  - all copyright rights + right to sublicense
  - all patent rights + right to “otherwise transfer” all Licensed Patents

“This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution.”

- applies to both source and object form

# EPL: 2. Grant of Rights

- Disclaimer of assurance and liability related to IP infringement:

“no assurances are provided...that the Program does not infringe the patent or other intellectual property rights...[and] disclaims any liability...for claims brought by any other entity based on infringement of intellectual property rights...if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

# EPL: 3. Requirements

- “Contributor may choose to distribute the Program in object code form under its own license agreement, provided that
  - it complies with the terms and conditions of this Agreement; and
  - the license effectively disclaims on behalf of all Contributors all warranties...
  - the license effectively excludes on behalf of all Contributors all liability...
  - the license states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
  - the license states that source code for the Program is available from such Contributor, and informs licensees how to obtain it...

# EPL 3. Requirements

- When distributed in source form, it must:
  - be made available under this Agreement
  - a copy of this Agreement must be included with each copy
- Each Contributor must identify itself as the originator of its Contribution

# EPL: 4. Commercial Distribution

“if a Contributor includes the Program in a commercial product offering, such Contributor...agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering...does not apply to any claims or Losses relating to any actual or alleged intellectual property infringement.”

# EPL: 5. No Warranty

- Unless required by law, Program is on a “AS IS” basis, including:
  - TITLE,
  - NON-INFRINGEMENT,
  - MERCHANTABILITY,
  - FITNESS FOR A PARTICULAR PURPOSE

# EPL: 6. Disclaimer of Liability

- Neither Recipient nor Contributors are liable for damages arising as a result of using or distributing the Program

# EPL: 7. General

- “If any provision of this Agreement is invalid...it shall not affect the validity... of the remainder of the...Agreement”
- “if Recipient institutes patent litigation against any entity... alleging that the Program itself...infringes such Recipient's patent(s) then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.”
- “All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time”

# EPL: 7. General

- “Everyone is permitted to copy and distribute copies of this Agreement, but...Agreement Steward reserves the right to publish new versions...after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version.”
- Governed by the laws of the State of New York

# GPL v2: Reminder

- Reach:
  - reciprocity covers container/derivative works
- Sublicense:
  - no clear sublicense grant; license granted from original author upon each distribution
- Patent Grant:
  - no explicit patent grant, but license implies one
- Retaliation:
  - no distribution unless everyone can distribute RF

# GPL v2 Highlights

- Section 0:
  - a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language.
- Section 1:
  - You may copy and distribute verbatim copies of the Program's source code
    - copyright notice and disclaimer of warranty;
    - keep intact all the notices and absence of any warranty;
    - give copy of License with Program

# GPL v2 Highlights

- Section 2:
  - b) 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.
- Section 3
  - You may copy and distribute the Program (or a work based on it, under Section 2) in object code, provided:
    - Accompany it with source code, or
    - written offer, or
    - information about the offer

# GPL v2 Highlights

- Section 4
  - You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License.

# GPL v2 Highlights

- Section 6

- Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions.

- Section 7

- If a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

# GPL v3 Highlights

- Section 9
  - If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

# AfferoGPL

- \* d) If the Program as you received it is intended to interact with users through a computer network and if, in the version you received, any user interacting with the Program was given the opportunity to request transmission to that user of the Program's complete source code, you must not remove that facility from your modified version of the Program or work based on the Program, and must offer an equivalent opportunity for all users interacting with your Program through a computer network to request immediate transmission by HTTP of the complete source code of your modified version or other derivative work.

# CeCILL v2 Highlights

- Two principles
  - Comply with Free Software principles
  - Conform to French Tort and IP Law
- Authors: CEA, CNRS, INRIA
- Preamble
  - Access to source
  - copy, modify, redistribute
  - certain obligations
  - limited warranty / limited liability
  - “risks associated with loading....”, “complicated to use”, “intended for use by computer professionals”

# CeCILL Highlights

- 1. Definitions
  - Holder: of economic rights
  - Licensee: users having accepted the agreement
  - Software: source and/or object code and/or docs
  - Modified Software: Software modified by at least one Contribution
  - Contribution: modifications and/or new functions integrated into the Software + Internal Modules
  - Module: sources files & their docs enabling supplementary functions or services
  - Internal Module: Module in same address space
  - External Module: Module in different address space and not derived from Software

# CeCILL Highlights

- 2. Purpose
  - grant by Licensor to Licensee
    - non-exclusive
    - transferable
    - worldwide
  - as set forth in Article 5

# CeCILL Highlights

- 3. Acceptance
  - “Licensee...deemed accepted the terms and conditions...upon...first of:”
    - “downloading from a remote server, or by loading from physical medium”
    - “exercises any of the rights granted hereunder”
  - Licensee hereby acknowledges that it has read and understood Agreement

# CeCILL Highlights

- 4. Effective Date and Term
  - effective when accepted
  - term = entire legal term of protection of the economic rights over Software

# CeCILL Highlights

- 5. Scope of Rights Granted
  - “if Licensor owns or comes to own one or more patents protecting all or part of the functions of the Software or of its components, the Licensor undertakes not to enforce the rights granted by these patents against successive Licensees using, exploiting, or modifying the Software”

# CeCILL Highlights

- 5.1 Right of Use
  - no limitation to field of application
  - right of reproduction: both permanent and temporary
  - load, display, run, store
  - observe, study, or test (reverse engineer)
- 5.2 Entitlement to Make Contributions
  - right to make Contributions
    - translate, adapt, arrange, or make any and all modifications
    - right to reproduce resulting Software
  - provided explicit notice of author and date of creation

# CeCILL Highlights

- 5.3 Right of Distribution
  - right to publish, transmit, and communicate to public
  - right to market for free or for a fee
  - may distribute source or object code, provided
    - a copy of the CeCILL is included
    - the warranty/liability notice requirements are observed (see Articles 8 & 9)
  - if only distributing object code, must indicate how to get the source
  - cost of source cannot exceed cost of transferring the data

# CeCILL Highlights

- 5.3 Right of Distribution
  - External Modules can be distributed under any license (CeCILL does not apply)
  - CeCILL is compatible with GPL
    - may include GPL (any version) code in CeCILL project and distribute entire code under GPL
    - may include CeCILL code in GPL (any version) project and distribute entire code under GPL

# CeCILL Highlights

- 6. Intellectual Property
  - Terms for Initial Software cannot be modified
  - Contributors still own their Contributions
  - Owners of an External Module can also license these modules as they like
  - Do not remove IP notices; reproduce existing notices in copies

# CeCILL Highlights

- 7. Related Services
  - No obligation to provide technical assistance
  - Only the one offering assistance shall incur liability for it.
  - Only the one offering warranty shall be bound by it.

# CeCILL Highlights

- 8. Liability
  - Licensee entitled to claim compensation for any direct loss (with evidence) as a result of a Licensor for commitments made in license, but NOT:
    - loss due to licensee not fulfilling obligations
    - direct loss due to use or performance of Software
    - any consequential loss (including financial/business loss and third-party suits)

# CeCILL Highlights

- 9. Warranty
  - all possible uses have not been tested
  - all possible defects/errors have not been detected
  - Software is reserved for experienced users, provided “as is”
  - Licensee shall verify suitability for their requirements, ensuring no damage to persons/property
  - Licensor represents it is entitled to grant all rights over Software,
    - but does not warrant that Software does not infringe any third party IP rights;
    - however, Licensor shall provide technical and legal assistance to help defend for any such litigation (see case-by-case MOU)
  - No warranty regarding trademark rights.

# CeCILL Highlights

- 10. Termination
  - Upon breach by Licensee, Licensor may terminate if not fixed in 30 days
  - However, previous distributions under this licenses shall remain valid.
- 11. Miscellaneous
  - No “failure to perform” liability attributable to “act of God”
  - Failure to enforce a provision is not a waiver for the next time.
  - Any modifications to terms must be in writing and signed.
  - Invalidity of part of license doesn’t invalidate the whole thing.
  - Both French and English versions are authentic.

# CeCILL Highlights

- 12. New Versions of the Agreement
  - permission to copy and distribute this license
  - only license authors may release new versions of license
  - Software licensed under one version can be distributed under the same or subsequent version

# CeCILL Highlights

- 13. Governing Law and Jurisdiction
  - governed by French law
  - if no amicable solution in two months, refer to Paris Courts

# LGPL v2: Reminder

- Reach:
  - covers containers/derivative works, but only restricts (does not require reciprocity) of licenses
- Sublicense:
  - no clear sublicense grant; license granted from original author upon each distribution
- Patent Grant:
  - no explicit patent grant, but license implies one
- Retaliation:
  - no distribution unless everyone can distribute RF

# LGPL v2: Section 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.

# LGPL v2: Section 5

- When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

# LGPL v2: Section 5

- If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)
- Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

# LGPL v2: Section 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.

# LGPL v2: Section 6

- 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:

# LGPL v2: Section 6

b) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

## LGPL v2: Section 6

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

# Java & LGPL FAQ

*The following Q&As are entirely the opinion of Cliff Schmidt and not necessarily that of the Free Software Foundation...*

*(and is not legal advice)*

# Java & LGPL FAQ

- Q1: If I create an original work of Java source code that imports, implements, or extends an LGPL-licensed Java class, are there any restrictions as to how I may license the source form of my original work?
  - A1: No - not unless there are portions of the LGPL-licensed work copied into your work, in which case the source would be classified as a derivative work or “a work based on the library”. If the work is truly original (uses only names and/or method signatures from the Library), the author may grant a license to any of the rights applicable to copyright ownership in their work.

# Java & LGPL FAQ

- Q2: If I compile the source code described in Q1, are there any restrictions as to how I may license the resulting Java class file?
  - A2: No - even though the normal Java compilation process may cause small portions (e.g. constant values/strings within a library's Java interface) of the LGPL-licensed work to be copied into the resulting class file(s), Section 5 of the LGPL states that there are no restrictions on the use of such resulting file. (“If such an object file uses only numerical parameters, data structure layouts and accessors ...then the use of the object file is unrestricted, regardless of whether it is legally a derivative work”)

# Java & LGPL FAQ

- Q3: If I take the source code from Q1 or the class file from Q2 and compile it with the LGPL-licensed work into native code, are there any restrictions on how I may license a distribution of the resulting executable?
  - A3: Yes - see Section 6 of the LGPL.

# Java & LGPL FAQ

- Q4: If I create a Java archive (e.g. JAR/WAR/EAR format) that includes both my original work from Q1 and the LGPL-licensed classes on which it depends, are there any restrictions as to how I may license the distribution of such an application or library?
  - A4: Section 7 of the LGPL requires that you do nothing to prohibit the separate distribution of the components of the archive (e.g. including a term in the archive license that disallows separate distribution). In addition, since you are distributing the LGPL work in object form (the classes/archive within the archive), Section 4 requires that you distribute the source in the same manner that you distribute the object code. See Sections 1 and 2 for more on distributing source - note that Section 1 also requires distributing the LGPL license with the source.

# Java & LGPL FAQ

- Q5: Wouldn't both the native code application described in Q3 and the Java archive application described in Q4 be considered “an executable”, and, therefore, subject to the requirements Section 6 puts on the terms of distribution of such an executable?
  - A5: No - a JAR simply bundles and compresses multiple files (using the common ZIP format).

# Java & LGPL FAQ

*(answer A5 continued)*

The mere aggregation of an LGPL-licensed library with other software inside a JAR does not bring the other software under the scope of the LGPL any more than would be the case by placing the same set of software in a ZIP, in a TAR, or on a CD-ROM. The fact that a Java Virtual Machine knows how to execute the components inside a JAR format does not disqualify it as a means of distributing an aggregation of software. However, Section 6 could apply to an inextricable combination of the LGPL-licensed library and other code, which may be considered a single derivative work; examples of such works would include C-style executables and Java code compiled into a native executable. The provisions of Section 6 force such typically inextricable executables to allow a user to replace the LGPL library with an improved version. The applicability of Section 6 isn't necessary for the distribution of the JAR described in Q4, since it is already inherently trivial for a user to accomplish what the LGPL wants to ensure.

# Java & LGPL FAQ

- Q6: It is still not clear to me whether my distribution, which includes an LGPL-licensed library, falls more under the terms of Sections 4, 6, or 7. What exactly is required of me to distribute this software?
  - A6: You should consult your own lawyer for your specific case and consider the entire LGPL. However, here is a summary of some of the LGPL's requirements for distributing the JAR described in Q4, regardless of which section one considers most applicable:
    - Include a copy of the LGPL with the distribution.
    - Include prominent notice that your software includes the LGPL-licensed library.

# Java & LGPL FAQ

*(answer A6 continued)*

- Distribute a copy of the source code for the LGPL-licensed library by the same means that you are distributing your software (e.g. same CD-ROM or web site).
  - do not alter any copyright, disclaimer, or license notices in the source.
  - see section 2 if you have made modifications to the LGPL-licensed library
  - include a pointer to this source in the prominent notice mentioned above.
- Ensure that you have not (through either technical or legal means) prohibited your users from replacing the LGPL-licensed library in the JAR with an improved (interface-compatible) version. If you are uncertain about this point, assume Section 6 applies.

# GPL v3: The Primary Differences from v2

- Clarifications over “Source Code”
- Patent Grant and Retaliation
- DRM
- License Compatibility

# GPL v3: Clarifications

- “The ‘Complete Corresponding Source Code; for a work in object code form means all the source code needed...For example...any shared libraries and dynamically linked subprograms that the work is designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work, and interface definition files associated with the program source files...”
- “As a special exception, the Complete Corresponding Source Code need not include a particular subunit if (a) the identical subunit is normally included as an adjunct in the distribution of either a major essential component...of the operating system...and (b) the subunit...serves only to enable use of the work with that...component...or to implement a widely used or standard interface, the implementation of which requires no patent license not already generally available...”

# GPL v3: Clarifications

- “A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, in or on a volume of a storage or distribution medium, is called an ‘aggregate’ if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. Mere inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.”

# GPL v3: Patent Grant and Retaliation

- “When you distribute a covered work, you grant a patent license to the recipient, and to anyone that receives any version of the work, permitting, for any and all versions of the covered work, all activities allowed or contemplated by this License, such as installing, running and distributing versions of the work, and using their output. This patent license is nonexclusive, royalty-free and worldwide, and covers all patent claims you control or have the right to sublicense, at the time you distribute the covered work or in the future, that would be infringed or violated by the covered work or any reasonably contemplated use of the covered work.

# GPL v3: Patent Grant and Retaliation

- If you distribute a covered work knowingly relying on a patent license, you must act to shield downstream users against the possible patent infringement claims from which your license protects you.”
- “This License gives unlimited permission to privately modify and run the Program, provided you do not bring suit for patent infringement against anyone for making, using or distributing their own works based on the Program.”

# GPL v3: DRM

- “Complete Corresponding Source Code also includes any encryption or authorization codes necessary to install and/or execute the source code of the work...”
- “No covered work constitutes part of an effective technological protection measure: that is to say, distribution of a covered work as part of a system to generate or access certain data constitutes general permission at least for development, distribution and use, under this License, of other software capable of accessing the same data.”

# GPL v3: License Compatibility

- “When you release a work based on the Program, you may include your own terms covering added parts...as long as those terms clearly permit all the activities that this License permits...those parts may be used separately under your terms...entire work remains under this License.”

# GPL v3: License Compatibility

- “Aside from additional permissions, your terms may add limited kinds of additional requirements on your added parts, as follows:
  - a) require preservation of certain...legal notices, and/or author attributions, ...and/or that altered versions of them be marked in the source code...
  - b) state a disclaimer of warranty and liability...different from...this License.
  - c) limit the use for publicity purposes of specified names of contributors, and...require that...trademarks be used...with express permission.
  - d) require that the work contain functioning facilities that allow users to immediately obtain copies of its Complete Corresponding Source Code.

# GPL v3: License Compatibility

- e) impose software patent retaliation, which means permission for use of your added parts terminates...conditions must limit retaliation to a subset of:
  1. Lawsuits that lack the justification of retaliating against other software patent lawsuits that lack such justification.
  2. Lawsuits that target part of this work...



coffee break



# Beyond the Fundamentals

Copyright 2006 Cliff Schmidt  
Licensed under [Creative Commons Attribution-NonCommercial-NoDerivs 2.5](https://creativecommons.org/licenses/by-nc-nd/2.5/)

ApacheCon  
Europe 06



# Software Interoperability

- Goal: to copy strictly only what is needed to achieve compatibility with other software.
- Remember:
  - idea-expression merger
  - scènes à faire
  - de minimus work
- For compatibility arguments, have your counsel review:
  - Computer Associates v. Altai: Three-Step Substantial Similarity Test
    - Abstraction-Filtration-Comparison test
    - Filters elements dictated by external factors including compatibility requirements, design standards, and industry demands.

# Case Law

- Lotus v. Borland (1995-96)
  - Menu hierarchies
    - organization of commands in menu structure
  - Methods of operation not copyrightable by Sect. 102
  - Specific words essential to operation
  - Menu command hierarchy & VCR buttons
    - command hierarchy isn't labels it's the button itself

# Case Law

- Midway v. Artic (1983)
  - add-on circuit board for video arcade games
  - caused game to run faster
  - contained no copied material
- Worlds of Wonder v. Veritel (1986)
  - Teddy Ruxpin teddy bear
  - cassette tapes derivative work causing Teddy to perform a similar act
  - issue of market effect

# Case Law

- Micro Star v. FormGen (1998)
  - map files containing no expression from the program can still be derivative
  - stories were sequels
- Sega v. Accolade
  - reverse engineering permitted
  - copying necessary parts for compatibility okay?
- Sony v. Connectix
  - reverse engineering permitted
  - copying necessary parts for compatibility okay?

# Case Law

- Galoob v. Nintendo
  - 9th Circuit on “derivative work”:
    - having "'form' or permanence”
    - "the infringing work must incorporate a portion of the copyrighted work in some form”

# Key Open Source Cases

- Welte v. Sitecom (2004)
  - distribution of netfileter/iptables without license and source
  - Munich court granted an injunction against Sitecom
- Wallace v. FSF (2006)
  - Sherman Act antitrust claims dismissed in US Federal Court
  - "[The GPL] acts as a means by which certain software may be copied, modified and redistributed without violating the software's copyright protection. As such, the GPL encourages, rather than discourages, free competition and the distribution of computer operating systems, the benefits of which directly pass to consumers. These benefits include lower prices, better access and more innovation."

# Fair Use

- Authorized uses of copyrighted works that would otherwise constitute infringement.
  - necessary to fulfill Constitution's mandate to "promote the Progress of Science and the useful Arts"
- Formally adopted in 1976 Copyright Act, although applied since 1800s.
  - law is still very unclear/subjective/uncertain
  - often a better risk/cost to obtain a license for required actions
- Four Factor Test...

# Fair Use: Four Factor Test

1. purpose and character of the work  
is it commercial or non-profit educational?
2. nature of the work  
fanciful or factual? published or unpublished?
3. substantiality of the portion used relative to whole  
measured both qualitatively and quantitatively
4. effect of use upon potential market for or value of work  
favors the defense when work has no economic value  
often considered most important factor

# Three Types of Copyrightable Works

- Creative
  - novel
- Derivative
  - screenplay based on novel
- Compiled
  - a fact book
- U.S. 11th Circuit in *Warren v. Microdos Data*:
  - most protection for creative works, least ('thin') for compiled

# Derivative Works Revisited

- Derivative works:
  - recast, adapt or transform an original work
  - while still remaining substantially similar
  - and may or may not need to be original (?)
- Examples:
  - Not derivative:
    - commentary “based upon” a work without copying its expression
    - insertion of commercials at start of video tape movie
  - Derivative:
    - sped-up version of a video game
    - removing 24 mins of a 90-minute Monty Python television special
    - Tivo’s 30-second skip?

# Trademarks

- Federal Trademark/Lanham Act of 1946 and state laws
  - part of unfair competition (e.g. false advertising, trade name confusion)
- Term of Protection:
  - perpetual, as long as it continues to be used
- Use It or Lose It
  - right to a particular mark grows out of its use, not its mere adoption (unlike copyright and patents)

# Trademark Strength

- Generic
  - not allowed trademark
- Descriptive
  - trademark after period of exclusive use (established secondary meaning)
- Suggestive
  - trademark immediately upon use
- Arbitrary/Fanciful
  - trademarked upon use

# Trademark Fair Use

- defendant's use is solely to describe defendant's goods, not as a trademark
  - not as a symbol to attract public attention
- freedom to use English language to describe goods
- questionable when likely confusion is established
- separate concept from comparative advertising

# Licensing a Trademark

- Important: licensor must control quality of goods sold under mark!
- Naked licensing -> abandonment
- Quality control can take many forms, but not simply reliance on licensee's quality control
- Best practice: Trademark License Agreement
  - validity of mark and licensor's right to control quality
  - specific quality standards or requirement for licensor approval/inspection
  - approval of advertising
  - not just a contract, but practiced

# IP Best Practices of OSS Organizations

Copyright 2006 Cliff Schmidt  
Licensed under [Creative Commons Attribution-NonCommercial-NoDerivs 2.5](https://creativecommons.org/licenses/by-nc-nd/2.5/)

ApacheCon  
Europe 06



# Eclipse Copyright and License Notices

- /\*\*\*\*\*  
\* Copyright (c) {DATE} {INITIAL COPYRIGHT OWNER} {OTHER COPYRIGHT OWNERS}.  
\* All rights reserved. This program and the accompanying materials  
\* are made available under the terms of the Eclipse Public License v1.0  
\* which accompanies this distribution, and is available at  
\* <http://www.eclipse.org/legal/epl-v10.html>  
\*  
\* Contributors:  
\* {INITIAL AUTHOR} - initial API and implementation  
\*\*\*\*\*/

# Apache Source File Header

0. This section refers only to works submitted directly to the ASF by the copyright owner or owner's agent.
1. If the source file is submitted with a copyright notice included in it, the copyright owner (or owner's agent) must either:
  - a. remove such notices, or
  - b. move them to the NOTICE file associated with each applicable project release, or
  - c. provide written permission for the ASF to make such removal or relocation of the notices.

# Apache License Notice

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

# Apache NOTICE file

- 0. Every Apache distribution should include a NOTICE file in the top directory, along with the standard LICENSE file
- 1. The top of each NOTICE file should include the following text, suitably modified to reflect the product name and year(s) of distribution of the current and past versions of the product:

Apache [PRODUCT\_NAME]

Copyright [yyyy] The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<http://www.apache.org/>).

# Apache NOTICE file

- 2. The remainder of the NOTICE file is to be used for required third-party notices. The NOTICE file may also include copyright notices moved from source files submitted to the ASF (see B.1.).

# Apache Notice Policy on Third-Party Work

0. The term "third-party work" refers to a work not submitted directly to the ASF by the copyright owner or owner's agent.
1. Do not modify or remove any copyright notices or licenses within third-party works.
2. Do not add the standard Apache License header to the top of any third-party source files.
3. Minor modifications/additions to third-party source files should typically be licensed under the same terms as the rest of the source for convenience.
4. Major modifications/additions to third-party should be dealt with on a case-by-case basis by the PMC.

# Copyright Assignment

- Imagine an exclusive copyright license for all rights and territories...
- In the U.S., transfer of copyright ownership must be in writing.
- In countries with moral rights laws, a complete waiver of ownership is not legal.
- Provides the most flexibility for future licensing options
- Provides the least incentive for outside contributions
- Often used with dual licensing strategies
- But, why must a broad copyright grant be exclusive? ....

# Exploring Licensing Practices of Apache & Eclipse

- Compare how Apache and Eclipse address these issues:
  - Copyright and Patent Licensing
  - Contributor Licensing Agreements
  - Handling Work-for-Hires
  - Distributing Third-Party Works

# Where Does the IP Come From?

- Contributors
  - minor contributions
  - significant contributions
- Committers
  - individuals
  - work-for-hire
- Inclusion of Third-party Components

# Contributions from “Contributors”

- Minor Contributions
  - Eclipse Terms of Use

“uploaded, submitted or otherwise made available to The Eclipse Foundation...you grant... the Eclipse Foundation...a worldwide, unrestricted, royalty free, fully paid up, irrevocable, perpetual, non-exclusive license to use, make, reproduce, prepare derivative works of, publicly display, publicly perform, transmit, sell, distribute, sublicense or otherwise transfer...”

- Apache License, v2

“*Contribution* intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.”

# Contributions from “Contributors”

- Significant Contributions
  - Eclipse Contributor Questionnaire (+ Terms of Use)
    - Contribution’s name, version, size
    - Contributor’s name, employer, email
    - Project Management Committee approval
  - Apache Software Grant (+ Apache License, v2)
    - signed statement by contributor with terms similar to the Eclipse Terms of Use grant shown on previous slide

# Contributions from “Committers”

- Individuals
  - Eclipse Individual Member Agreement

“Committer agrees that the EPL...as set out in the Terms of Use will serve as...the general contribution license”
  - Apache Individual Contributor License Agreement

“You accept and agree to the following terms and conditions for Your present and future Contributions submitted to the Foundation...You hereby grant to the Foundation and to recipients of software distributed by the Foundation a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright [and patent] license to...”

# Apache's Contributor License Agreement

- It's a license - not copyright assignment!
  - “You reserve all right, title, and interest in and to your Contributions.”
- Applies to all present and future contributions
- Grants copyright license to Foundation and recipients
  - same rights as granted to users in the Apache License v2.0
- Grants a patent license to Foundation and recipients
  - same rights as granted to users in the Apache License v2.0, including the patent termination clause
- Legally entitled to grant license (see cCLA)
  - Submission must disclose any known restrictions (e.g. patent/trademark licenses) - update if become aware of inaccuracy
- No warranty being made by contributor

# Contributions from “Committers”

- Eclipse Work-for-Hire (employees)
    - Eclipse Member Committer Agreement
      - references named employees of member company
  - OR
  - Eclipse Employer Consent Form
    - “Do you agree that Employee’s current and future contributions, whether such contributions are owned by employer or employee, will be provided under...the [EPL]?”
- 
- Apache Work-for-Hire (employees)
    - Apache Corporate Contributor License Agreement
      - covers named employees, who must still sign Individual CLA

# Inclusion of Third-Party Components

- Eclipse Procedures

“Before any such package can be redistributed by the Eclipse Foundation, the Committer must contact the PMC for a decision on whether the package’s functionality is required and accepted for the project. Once the PMC approves the contribution, the PMC must contact the EMO to...obtain approval... regarding the compatibility of the contribution’s license with the EPL, and...initiate the due diligence review by the EMO.”

# Third-Party License Policy

see <http://www.apache.org/~cliffs/3party.html>

# How Does the IP Get Licensed?

- Eclipse Foundation Software User Agreement

*“Unless otherwise indicated, all Content made available by the Eclipse Foundation is provided to you under the terms and conditions of the [EPL]...THE ABOUTS...MAY REFER TO THE EPL OR OTHER LICENSE AGREEMENTS, NOTICES OR TERMS AND CONDITIONS. SOME OF THESE OTHER LICENSE AGREEMENTS MAY INCLUDE (BUT ARE NOT LIMITED TO):*

*\* Common Public License Version 1.0...*

*\* Apache Software License 2.0...*

*\* Mozilla Public License Version 1.1....*

*IT IS YOUR OBLIGATION TO READ AND ACCEPT ALL SUCH TERMS AND CONDITIONS PRIOR TO USE OF THE CONTENT. ”*

# How Does the IP Get Licensed?

- Apache's LICENSE.txt file:

*“This product is a collective work, selected and arranged by the Apache Software Foundation and licensed under the Apache License, Version 2.0. The components of this work include:a) software licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements, which the ASF licenses to you under the Apache License, Version 2.0; and possibly,b) third-party components licensed under terms that may be different from those of the Apache License, Version 2.0. Copies of such licenses can be found in either a file or directory labeled '3PARTY\_LICENSES'.”*

# IP Due Diligence Review

- Typical IP review tasks
  - ensure package license represents all included code
  - keyword search for other copyright notices, etc
  - license compatibility checks
  - investigation of any suspect work-for-hire issues
  - research into procedures for contributions/distribution
  - discussion of concerns with authors or applicable organizations (other open source orgs, employers)

# Handling Patent Concerns and Infringement Claims

- Standards and Patent Licenses
- Handling a Patent Infringement Claim

# Interpreting combinations of contribution to work

- Breadth of definition of Work
- Work now and in the future (with future changes)
- Patents held now or in the future

# Trademark Management

- Goal: control connotation of source and quality
- Example: Guidelines for Eclipse Logos & Trademarks
  - “Derivative works of the EclipseCon, Eclipse Foundation Member, Built on Eclipse and Eclipse Ready logo are not allowed.”
  - “You may not incorporate the Eclipse trademark into the name of your company or software product name. If you have a software product that works with Eclipse, it is suggested you use terms such as ‘<product name> for Eclipse’ or ‘<product name>, Eclipse Edition’”

# Trademark Management

- “The "Built on Eclipse" trademark is used to identify a product that includes the core Eclipse Platform, specifically the runtime, SWT, JFace, and Generic Workbench components. This type of product is able to run standalone with no prerequisites of other Eclipse components.”
- “The Eclipse Incubation and Eclipse Proposals Trademarks and Logos are for use exclusively by the Eclipse Foundation. Only the Eclipse Foundation is authorized to use these trademarks and logos. All other use -- even by Eclipse Foundation Members -- is prohibited.”

# Questions?

