



Lecture 25: Licensing and legal considerations

CS 5150, Spring 2022

Course reminders

- Final presentations during this session
- Final delivery May 10
 - Requirements on course website
- Assignment A4 due Tue, May 3

Lecture goals

- Navigate the spectrum of open-source [licenses](#)
- Avoid pitfalls of software [patents](#)
- Be aware of applicable [regulations](#) and [export restrictions](#)

Legal context

Law

- Law is not code or pure logic
 - Definitions are ambiguous
 - Implications may be inconsistent
 - Consequences are not automatic
 - Interpreted by humans
- Law is not uniform
 - Varies by jurisdiction
 - Changes over time
 - Interpretation varies by judge
- Law is serious
 - Penalties include \$M-\$B fines, destruction of inventory, jail time
 - Court cases are slow and *expensive*
- Do not try to interpret the law on your own
 - Might be reading the wrong law
 - May not be aware of case precedent
 - People on the Internet can be wrong (gasp!)
- Most companies retain legal counsel – take advantage of them

IANAL; this is not legal advice

Jurisdiction

- The United States follows **Common Law**, composed of:
 - **Statutes** (bills) passed by Congress and states
 - **Regulations** issued by government
 - **Precedents** (judgements) made by courts
- Federal law and state laws, each with their own multi-layered court systems
- International law is complex and dynamic, especially regarding Internet services
 - A user, the server that they access, the server where their data is stored, and the headquarters of the company running the service may all be in different countries. Each country may claim jurisdiction over the service.

Computing-specific laws

- Computer Fraud and Abuse Act (1986)
 - Explicitly prohibits unauthorized access of computing systems
- Digital Millennium Copyright Act (DMCA, 1998)
 - Prohibits circumvention of access control to copyrighted works
 - Limits liability of service providers for actions by their users
- General Data Protection Regulation (GDPR, 2016), California Consumer Privacy Act (2020)
 - Restricts use of personal data – requires consent or legitimate interest
 - Browser cookies qualify as personal data

Other laws affecting software

- Contracts
- Copyright
- Patents
- Trade secrets
- Export
- Privacy

Software licensing

Copyright

- Broad international consensus thanks to Berne Convention
 - Applies to "expression of ideas"
 - Originally literary and artistic works
 - Established automatically when work is created
 - Initially held by creator *except* for [work for hire](#)
 - Can be transferred
 - Rights can be licensed
- Holder has exclusive right to:
 - Reproduce
 - Use as basis for derived works
 - Distribute
 - Exhibit in public
 - License rights to others
 - Exceptions
 - First sale: objects containing copyrighted work can be resold
 - Fair use: limited use is allowed without permission for special purposes (review, quotation)

Joint authorship

- Each coauthor can independently exercise their copyright rights (including licensing to third parties) without permission of other coauthors
- But by default, coauthors must share all profits equally with one another
- CS 5150 project contributions probably qualify as joint authorship
 - But is also protected by FERPA

Work for hire

- When working for an employer, you often assign all patent and copyright rights to them
 - Will need employer's permission before contributing to open-source projects
 - Can be negotiated at time of employment
 - Can ask for a copyright disclaimer to protect against future claims
- If doing freelance work, **must have a contract** with the client that declares ownership of copyright
- If you do not hold the copyright, you need explicit permission to copy or use software, *even if you wrote it yourself* (retaining a backup for personal use is not automatically allowed)

Contract and copyright law

- Most commercial software licenses fall under contract law
 - Software is not purchased for ownership; it is licensed for use
 - End User License Agreements (EULAs) are an alternative to signed contracts
- Contracts also govern software development work
 - Who will own copyright
 - Exclusivity of licensing
 - Contingency terms
- Most open-source licenses leverage copyright law

Open-source software licenses

- "Free as in beer" (gratis)
 - Users do not need to pay money to use the program
- "Free as in speech" (libre)
 - Users of software should be guaranteed freedoms, such as:
 - Free to use the software for any purpose
 - Free to read and modify source code
 - Free to distribute copies
- Two underlying philosophies:
 1. An open development model leads to better software ("open source" movement)
 2. Software should not violate users' freedoms (Free Software Foundation)



Permissive and copyleft licenses



Copyleft

- If copyleft-licensed software is **modified** and **redistributed**, then the redistributed software must be under the same (or a compatible) license
 - In particular, source code must be made available
 - Not a concern for internal use – if software is not released to the public, source code may remain private
- Epitomized by the GNU General Public License (GPL)
 - The GPL considers *linking* (both static and dynamic) to constitute a "derived work"
 - Often considered "viral"
- Other examples:
 - Mozilla Public License (file scope; less viral)
 - Creative Commons Share-Alike

GPL variations

- Lesser GPL (LGPL)
 - Applications linking to an LGPL library do not have to be distributed under the same license
 - Wording is C-centric
- GPL with classpath exception
 - Also removes virality, but more language-agnostic
 - Intended for runtime support and standard libraries
- Affero GPL (AGPL)
 - Providing network services linked with AGPL software requires making the server's source code available
- GPLv2
 - Avoids GPLv3 scope creep, but also its improved compatibility
 - Used by Linux

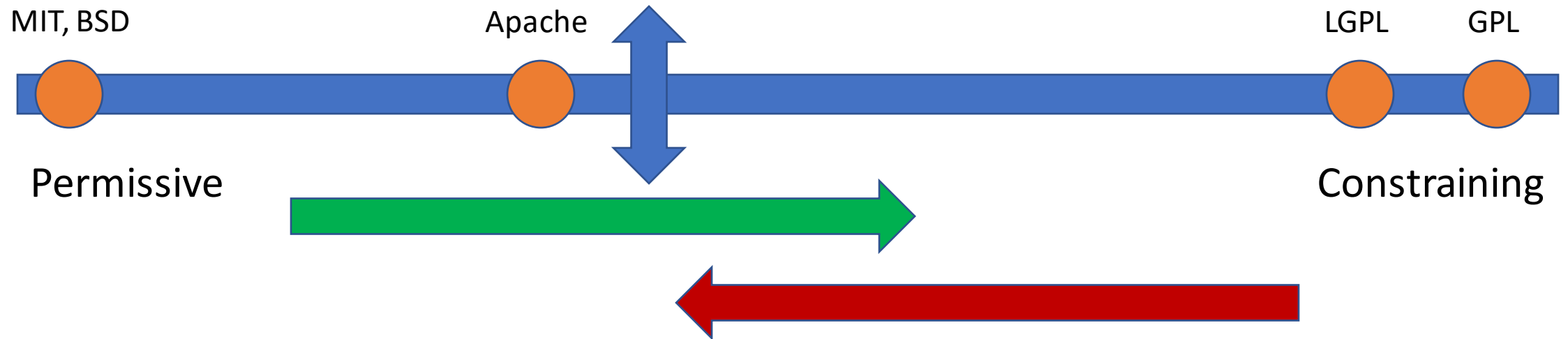
Other constraints

- Advertising clause
 - In original BSD license; not scalable
- Patent rights
 - Recipients of the software are granted rights to applicable patents as required for running the software
 - Patent rights revoked if you claim patent infringement
- Trademarks and non-endorsement
 - May not be necessary to state explicitly
- Limited liability & lack of warranty
- Attribution
 - Preservation of license and copyright notices

Examples

- BSD, MIT/X11/Expat, ISC
 - Simple, highly-permissive license
 - Many variations, functionally equivalent
- Apache License 2.0
 - Permissive
 - Patent rights & retaliation
 - Preferred by corporate contributors to open-source software (lawyer-friendly)
- GPL family
 - Copyleft

License compatibility



Licensing contributions

- "Inbound=outbound"
 - Contributions to an open-source project are implicitly licensed under that project's LICENSE at the time of commit (explicit on GitHub)
 - Copyright is held by contributors
- Contributor License Agreements
 - Assert contributors' right and intent to contribute
 - Potentially assign copyright to project stewardship
 - Potentially grant project steward right to relicense/dual license
 - Must be signed before any contribution can be accepted
- [Developer Certificate of Origin](#)
 - Lightweight agreement acknowledged in commit message ("Signed-Off-By:")

Applying a license

- Add a LICENSE file to the root of your repository
- Add a copyright and license notice header to every file
 - Not always required; see license documentation for recommendation
- Track licenses of dependencies, contributions with [SPDX](#) identifiers
 - Facilitates assembling a "software bill of materials" (licenses and copyrights of all included components)
 - When creating a software product, you must have *documented* rights to use everything from which it was derived
 - Example: Python

Licensing CS 5150 projects

- As Cornell students you retain copyright on code you write for coursework
- **External projects** should have signed a license agreement with their client during session 1 or 2
 - Must at least grant comprehensive usage license (including permission to create and use derived works) to client
 - May transfer copyright to client
 - Agreement must be in writing and signed by all contributors
- **Internal projects** must agree on how contributions should be licensed
 - Option 1: Agree to publish under inbound=outbound (can share on public GitHub, include in portfolio, contribute upstream)
 - Option 2: Treat as coursework. Cannot publish if students are identifiable

Resources

- <https://opensource.guide/legal/>
- <https://choosealicense.com/appendix/>
- <https://www.gnu.org/licenses/license-list.html> (biased, but thorough)

Patents and trade secrets

Patents

- Defines rights of inventors
 - Applies to ideas (whereas Copyright applies to an *expression* of ideas)
- Should be non-obvious, novel, and useful
- Patent rights
 - Exclude others from making, using, selling, or importing the invention
- Must submit an application to be granted a patent
 - Public disclosure in exchange for exclusive rights
 - Patents last 20 years from filing date
- Once granted, patent validity can be challenged in court
 - Extremely expensive; high burden of proof
 - Most patent disagreements are settled out of court

Software patents

- Problems

- Collaborative and rapidly-advancing culture – difficult to know where ideas originate
- Poor standards for examination leads to granting of broad patents for routine concepts
 - Some companies require filing patents for promotion
- Broad patents may overlap – hundreds may cover the same idea
- International differences

- Patent abuse

- Large companies hoard patents
 - Can harass smaller competitors
 - Used as defense against other large companies
- Patent trolls – business model is to collect unenforced patents and threaten litigation
 - Often harass smaller companies and end users
 - Large companies may pursue invalidation in court (Cloudflare Project Jengo)

If you receive letters threatening litigation, consult a lawyer

Software patent examples

- Arithmetic coding
 - Alternative to Huffman coding in JPEG format
 - Reduces file size by up to 25%
 - Covered by IBM patents until ~2010
 - Very few JPEG libraries support arithmetic coding; not used on Internet
- H.264 (MPEG-4 AVC)
 - MPEG LA maintains patent pool for most (but not all) required patents – over 6000
 - Patents cover not just encoding and decoding, but also transmission
 - Cisco provides a free binary codec for which they pay licensing fees
 - But rebuilding it from source would require you to pay your own fees

Trade secrets

- Companies can protect confidential business information
- Owner must make reasonable effort to maintain secrecy
- Employees may not disclose trade secrets, competitors may not use secrets obtained via extraordinary means
 - Leaks do not invalidate trade secrets
 - Protections never expire
- Non-disclosure agreements (NDAs)
 - Legal agreement to not disclose trade secrets
 - Common precondition for employment or even interviewing
 - But still read carefully

Export law

- Governments restrict what can be exported, even "intellectual property"
- Common export regulations affecting US software:
 - International Traffic in Arms Regulations (ITAR)
 - Export Administration Regulations (EAR)
- Intent: prevent export of goods and information that could assist adversaries in harming national interests
 - Applies to many aerospace technologies
 - Formerly applied to cryptography
 - Unintended consequence: compatibility with "export grade" cryptography makes services very insecure
- Penalties are *harsh* and apply to employees as well as companies

Privacy

- Computer systems often handle information that people would consider private
- Questions
 - What information can you collect? (legally and ethically)
 - How should you store private information?
 - Where can you store private information?
 - Who can you disclose private information to?
 - When should/must private information be destroyed?
- Laws and social norms are changing rapidly

Types of information

- Specific facts
 - Person A used Google to search for keyword B
- Metadata
 - An e-mail was sent from address A to address B
- Aggregate information (individuals could be identified from intersection of non-identifying facts)
 - A person is married, lives on Highland Rd, has a red Toyota, and works for Cornell

Common privacy regulations

- FERPA
 - Protects educational records of students
 - Grades should be private to individuals
 - Enrollment, work should be private to classroom
- HIPPA
 - Protects medical information
- Often need special tiers of third-party services (e.g. AWS, Zoom) to be compliant