

Intellectual Property

Lecture 4-2

Computers & Society (CPSC 430)

Kevin Leyton-Brown

<https://www.cs.ubc.ca/~kevinlb/teaching/cs430>

Intellectual Property Protection

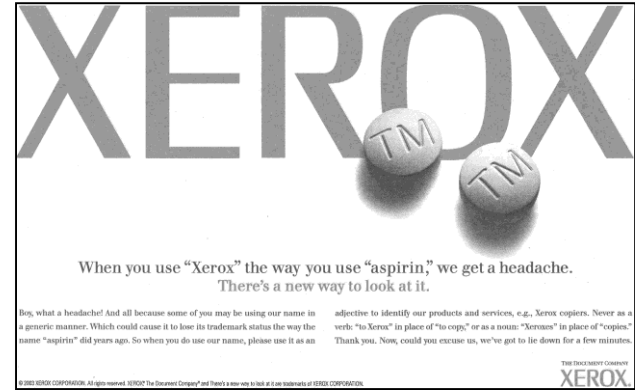
- **Summary: why Intellectual Property Protection?**
 - Some people are altruistic; some are not
 - Allure of wealth can be an incentive for speculative work
 - Thus, benefits to intellectual property protection
- **But, these rights should come with limits**
 - Giving creators rights to their inventions stimulates creativity
 - Society benefits most when inventions in public domain
 - Strike a compromise by giving authors and inventors rights for a limited time
- **It might make more sense to call it “intellectual monopoly” rather than “intellectual property”**

1. Trade Secret

- Confidential piece of intellectual property that gives company a competitive advantage
- Never expires
- No legal protection
- Reverse engineering allowed
- May be compromised when employees leave firm

2. Trademark, Service Mark

- Trademark: Identifies goods
- Service mark: Identifies services
- Company can establish “brand name”
- Does not expire
- If brand name becomes common noun, trademark may be lost
- Companies advertise to protect their trademarks
- Companies also protect trademarks by contacting those who misuse them



3. Patent

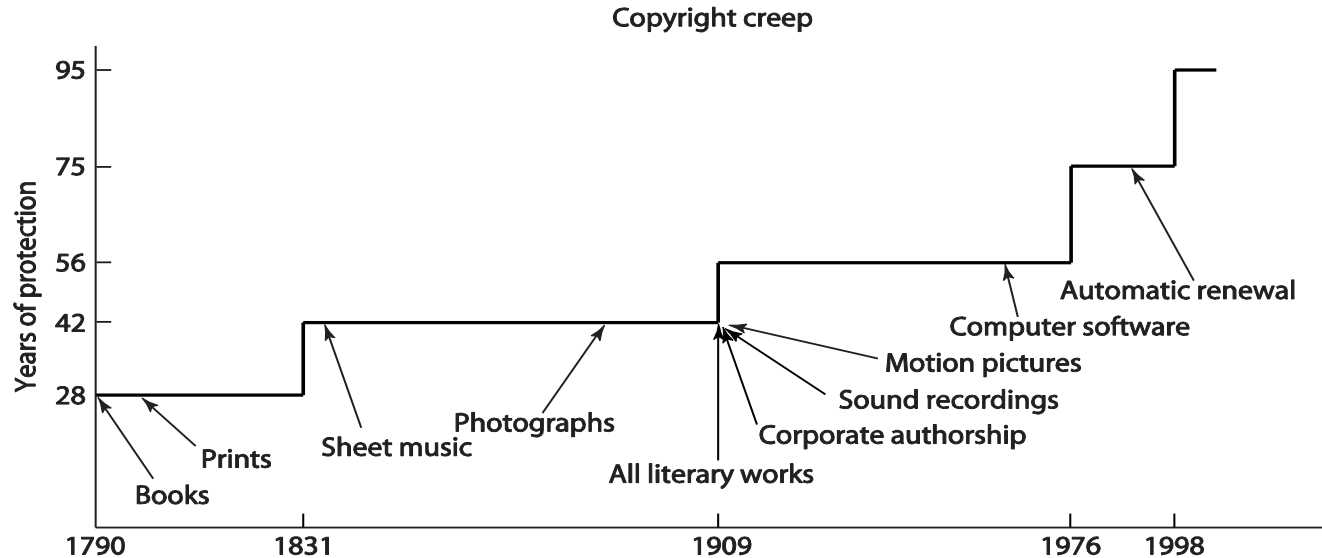
- A public document that provides detailed description of invention
- A government office decides whether the invention is novel, non-obvious
- Provides owner with exclusive right to the invention
- Owner can prevent others from making, using, or selling invention for 20 years

Software Patents

- Patent protection began in 1981
- Inventions can be patented, but not algorithms
- Patent Office having a hard time determining prior art
- Result: some bad patents have been issued
 - Amazon One-Click purchasing
 - Apple: squares with rounded corners
- General skepticism about value of software patents
- *Patent trolls: what are they? What do you think?*

Copyright Creep

- Since 1790, protection for books extended from 28 years to 95 years or more
 - latest extension aims to protect Disney characters from entering public domain?
- Copyright Term Extension Act of 1998 challenged as unconstitutional
- U.S. Supreme Court disagreed: CTEA doesn't create perpetual copyrights



Digital Rights Management

- Actions owners of IP take to protect their rights
- Approaches
 - Encrypt digital content
 - Digital marking so devices can recognize content as copy-protected
- **Example: the (failed) Secure Digital Music Initiative (2000)**
 - Consortium didn't stick together
 - Cracked by CS researchers
- **Example: Sony BMG Rootkit (2005)**
 - Made everyone angry; retracted
- **Example: online music stores (2003—2009)**
 - Started out with DRM, in part to lock people into platforms
 - Lately, moving away from it
- **Example: streaming sites (this decade)**

Fair Use/Fair Dealing

USA: Fair Use

- Cases where copyrighted work can be reproduced without permission
- Use can be for any purpose
- Usage must be fair
 - Purpose, character of use
 - Nature of work
 - Amount of work copied
 - Effect on market for work

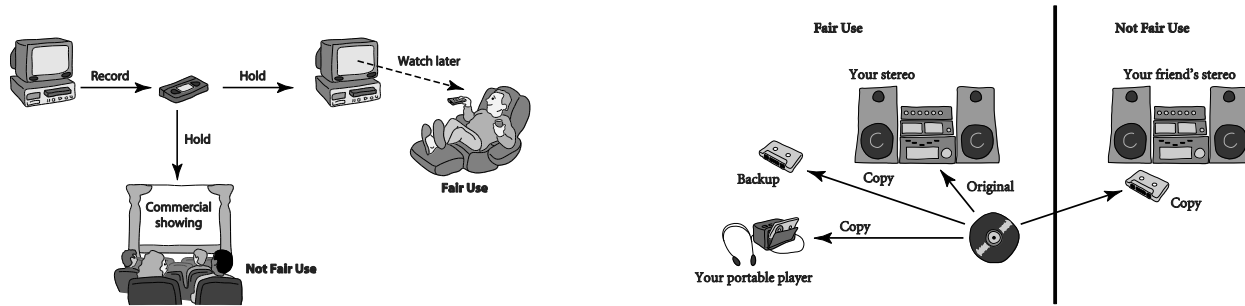
Canada: Fair Dealing

- Three protected activities:
 - research or private study
 - criticism or review
 - news reporting
- Usage must be fair
 - purpose (commercial/private)
 - character (e.g., # copies made)
 - amount copied from the original
 - alternatives (was copying necessary?)
 - nature (e.g., public availability of copyrighted work)
 - effect (does copy compete with original?)

Is it ethical to break a digital lock in order to make fair use of a copyrighted work?

Some famous examples of fair use

- “Time shifting” (recording shows on VCR/DVR)
- “Space shifting” (transcoding music onto MP3 player)



- Use of image thumbnails in search results
- Google books – indexing full texts

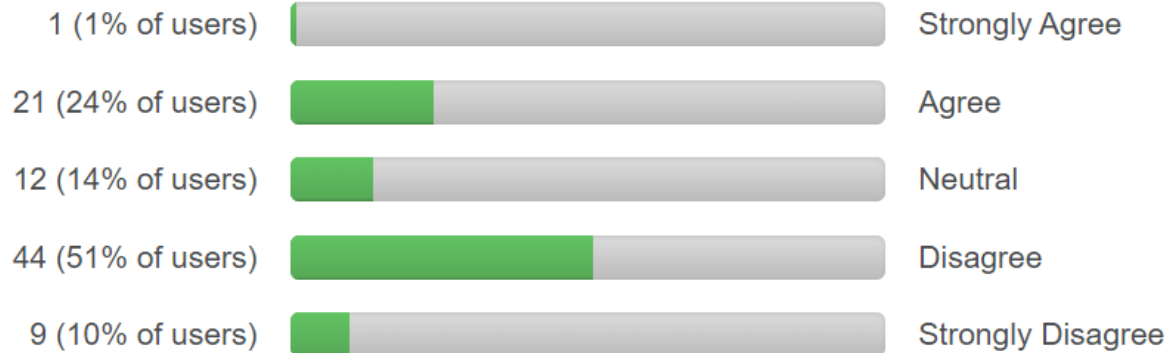
Legitimacy of IP Protection for Software

- Software licenses typically prohibit you from making copies of software to sell or give away
 - Our focus is not on whether it's ethical to violate such a legal agreement after having agreed to it.
- Instead, we are considering:
 - whether society *should* give IP protection to software
 - if so, how this protection ought to be limited
 - what ethical argument can be used to justify this protection.

Intellectual Property

“The government should aggressively prosecute intellectual property infringement, particularly including peer-to-peer file sharing.”

A total of 87 vote(s) in 75 hours



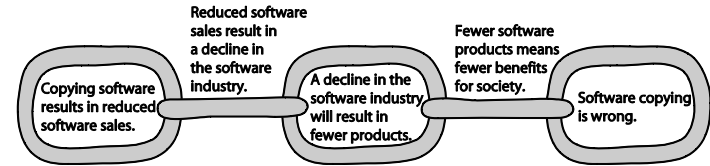
Rights-based Analysis

- “Just deserts” argument
 - Programming is hard work that only a few can do
 - Programmers should be rewarded for their labor
 - Mixing my labor with something implies ownership
- Criticism of “just deserts” argument
 - Why does labor imply ownership?
 - Maybe mixing my labor with something means I lose my labor
 - Pour a can of tomato juice into the ocean: I don’t own the ocean
 - A society in which all labor went to common good could be just
 - Intellectual property not like physical property
 - I cut logs: I own the logs
 - I write a book: I get to restrict other people from copying the book
- *What do you think about this argument?*

Utilitarian Analysis

- Argument against copying

- Copying software reduces software purchases...
- Leading to fewer software producers...
- Leading to lower production of new software...
- Leading to fewer benefits to society



- Each of these claims can be debated

- Not all who get free copies can afford to buy software
- Open-source movement demonstrates many people are willing to donate their software-writing skills
- Hardware industry wants to stimulate software industry; freemium model; many apps are supported via ads
- Difficult to quantify how much society would be harmed if certain software packages weren't released

- *What do you think about this argument?*

Legal Action Against P2P

- **RIAA Lawsuits (2003)**
 - Sued 100s of high-volume sharers
- **Universities hotbed for sharing**
 - Responses: banning, signing site licenses
- **MGM vs. Grokster**
 - Grokster won at lower levels, eventually lost at Supreme Court
 - Ruling: the technology existed primarily for infringement
- **Pirate Bay:**
 - Repeatedly shut down, sued, operators fined and jailed (2013-15), but it's still up

Open Source

- **A variety of licenses. Some typical ingredients:**
 - No restrictions preventing others from selling or giving away software
 - Source code included in distribution
 - No restrictions preventing others from modifying source code
 - No restrictions regarding how people can use software
 - Same rights apply to everyone receiving redistributions of the software (copyleft)
- **GNU Project (Richard Stallman, 1984-)**
 - Goal: Develop open-source, Unix-like operating system
 - Most components developed in late 1980s
- **Linux**
 - Linus Torvalds wrote Unix-like kernel in 1991
 - Combined with GNU components to make an OS
 - putting pressure on Microsoft, Apple, and companies selling proprietary versions of Unix

Benefits and Drawbacks of Open Source

- **Benefits**

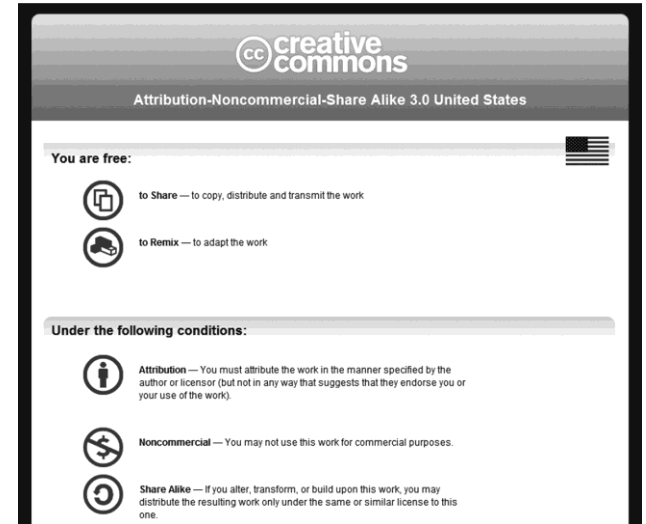
- Gives everyone opportunity to improve program
- New versions of programs appear more frequently
- Eliminates tension between obeying law and helping others
- Programs belong to entire community
- Shifts focus from manufacturing to service

- **Drawbacks**

- Without critical mass of developers, quality can be poor
- Without an “owner,” incompatible versions can arise
- Relatively weak graphical user interfaces
- Poor mechanism for stimulating innovation (no companies will spend billions on new programs)

Creative Commons

- Under current copyright law, eligible works are copyrighted the moment they are created
- No copyright notice does not mean it's ok to copy
- Must contact people before using work
- That slows down creative reuse
- Free Creative Commons license indicates
 - Which kinds of copying are ok
 - Which rights are being retained
- Flickr and Magnatune two well-known sites using Creative Commons licenses



Safe Software Development

- Reverse engineering okay
- Companies must protect against unconscious copying
- Solution: “clean room” software development strategy
 - Team 1 analyzes competitor’s program and writes specification
 - Team 2 uses specification to develop software
- Interestingly, same development strategies also used to ensure that open source licenses don’t “infect” commercial software