

# Intellectual Property

Readings: Tavani, Chapter 8: “Intellectual Property Disputes in Cyberspace”

- “Copyright, Designs and Patents Act 1988 (CDPA 1998): [http://  
www.legislation.gov.uk/ukpga/1988/48/contents](http://www.legislation.gov.uk/ukpga/1988/48/contents)
- “The Copyright and Rights in Database Regulations 1997” [http://  
www.legislation.gov.uk/uksi/1997/3032/made](http://www.legislation.gov.uk/uksi/1997/3032/made)

# Intellectual Property in Cyberspace

- Possibly **the** defining ethical issue of the digital era.
- The most contentious property issue in the past decade has probably been...
  - should users be able to freely download and exchange copyrighted music in the form of MP3 files?
  - ...or games! ... or software! ...or pictures!

# In the here and now for you:

- Do students own the software they produce as part of a university course?
  - E.g. for the final year project?
  - can they publish it (e.g. github?)
  - can they expand it and make profit out of it?

# Why Property Laws are Important

- Social scientists suggest that property laws play a key role in (a) shaping a society and (b) preserving its order by establishing relationships between:
  - individuals,
  - different sorts of objects,
  - the State.

# What Is (Tangible) Property?

- When discussing property, we tend to (used to?) think of tangible items.
- Originally, "property" referred to land.
- Property now also includes objects that one can own, such as:
  - a car,
  - articles of clothing,
  - a DVD collection.

# Property as a “Relational” Concept

- Property should not be viewed simply in terms of items or things (tangible or otherwise).
- Philosophers and legal theorists point out that property can best be understood as a relationship between individuals in reference to things, where three elements need to be considered:
  - 1) an individual (X),
  - 2) a “thing” or object (Y),
  - 3) X's relation to other individuals (A, B, C, etc.) in reference to Y.

# Property as a Form of “Control”

- X (as the owner of property Y) can control Y relative to persons A, B, C, and so forth.
- If a person owns a certain object (e.g. a laptop), then this person can control who has access to that object and how it is used.
  - For example, this person has the right to exclude another person from using that laptop; or could grant unlimited access to it.
- Ownership claims involving “intellectual objects” (involving IP) are both similar to and different from ownership of tangible objects.



# Intellectual Objects

- The expression *intellectual object* can refer to various forms, or instances of intellectual property.
- Intellectual property consists of “objects” that are not tangible.
- Intellectual objects represent creative works and inventions, i.e., the manifestations or expressions of ideas.



# Intellectual vs. Tangible

- Tangible objects are exclusionary in nature.
  - If a person owns a specific physical object, then another person cannot, and vice versa.
- Intellectual objects, such as software programs, are non-exclusionary.
  - If a person makes a copy of a word-processing program that resides in another person's computer, then both persons can possess copies of the same word-processing program.

# Intellectual vs. Tangible

- The sense of scarcity that applies to tangible objects, which often causes competition and rivalry, need not exist for intellectual objects.
  - For example, there are practical limitations to the number of physical objects that one can own.
  - There are also limitations (natural and political) to the amount of land that can be owned.
- Intellectual objects can be easily reproduced.
  - Countless copies of a software program can be produced – each at a relatively low cost.

# Ownership of Intellectual vs. Tangible Objects

- Legally, one cannot own an idea in the same sense that one can own a physical object.
- Governments do not grant ownership rights to individuals for ideas per se.
- Legal protection is given only to the tangible expression of an idea that is creative or original.

# Ideas vs. Expressions of Ideas

- If an idea is literary or artistic in nature, it must be expressed (or "fixed") in some tangible medium in order to be protected.
- A "tangible medium" could be a physical book or a sheet of paper containing a musical score.
- If the idea is functional in nature, such as an invention, it must be expressed in terms of a machine or a process.
- Authors are granted copyright protections for expressions of their literary ideas, while inventors are given patent protection for their inventions.

# Why Protect Intellectual Property?

- ... because the law says so!
- But: On what philosophical grounds are property laws themselves based?
- One can view property right as a "natural right," to which individuals are justified for the products that result from their labour, including intellectual objects.
- Or one can view property rights as a "social construct" designed to encourage creators and inventors to bring their artistic works and inventions into the marketplace.

# Philosophical Foundations for Intellectual Property

- Three distinct types of traditional (philosophical) theories regarding property rights can be articulated, i.e., the:
  - 1) labour theory;
  - 2) utilitarian theory;
  - 3) personality theory.
- It is important to note that some theorists argue against property protection of any kind.
- Others argue against protecting intellectual property, but not against tangible property.



# Property: Labour Theory

- Introduced by John Locke (17<sup>th</sup> century).
- According to the labour theory, a person is entitled to the results of his or her labour.
- Property rights are justified because of the one's physical "sweat of the brow."
- It makes definitely sense for physical objects, but one can see it applying to intellectual objects as well



# Property: Utilitarian Theory

- A property right is not a natural right; rather it is a conventional right granted by governments.
- Property rights should be granted because they provide an incentive to bring ideas into the marketplace (i.e. they have social utility).
- This theory (questionably) assumes that there must be an economic incentive to produce creative works; otherwise, creative works would not be produced.
- Utilitarian theory in general favours the interests of the greatest number of persons (affected by a social policy) at the expense of the minority population that is also affected by a social policy.

# Property: Personality Theory

- Traced back to G.W.F. Hegel (19<sup>th</sup> century).
- Property rights are not tied to labour, nor to economic incentives.
- A property right should be granted because of the personality of the author that is invested in the creative work (regardless of economic considerations).
- it considers that the use of a creative work may not represent the author's personality



## **Queen's Brian May Denies Donald Trump Usage of Band's Music**

"I can confirm that permission was neither sought nor given," guitarist says after Trump used song in victory speech

# Summary of the Three Philosophical Theories of Property

Labour Theory	Argues that a property right is a "natural right" and that property rights can be justified by the labour or "toil" that one invests in cultivating land or in creating a work of art.
Utilitarian Theory	Argues that property rights are not natural rights but rather artificial rights created by the state. Property rights are granted to individuals and to corporations because they result in greater social utility overall.
Personality Theory	Argues that a property right is a "moral right" and that property rights are justified not because of labor or social utility but because creative works express the personalities of the authors that create them.

# Software as Intellectual Property

- Should computer programs be eligible for property law protection? Of which sort? And, what is a computer program anyway?
- Computer software consist of lines of programming code
  - It is not expressed or "fixed" in a tangible medium in a way that literary works are.
- Its object code is made up of "executable images" that run on the computer's hardware after they have been converted from the original source code.
- But software programs also can be seen as the algorithm, like mathematical ideas or "mental steps".



# The Case for Protecting Software as a form of Intellectual Property

- The software industry has made the following kind of argument for why software should be protected with intellectual property rights.
  - **PREMISE 1.** Stealing a tangible object is morally wrong.
  - **PREMISE 2.** Making an unauthorised copy of proprietary software is identical to stealing a tangible object.
  - **CONCLUSION.** Making unauthorised copies of proprietary software is morally wrong.

# The argument for Protecting Software

- Premise 1 is fairly straightforward
- But Premise 2 is more controversial and thus we can question whether it is empirically true.
  - For example, is duplicating a software program identical to stealing a physical item?
    - Consider that software programs, like other intellectual objects, are non-exclusionary; so my having a copy of Program X does not exclude your having a copy of that same program, and vice versa.
- Because the truth of Premise 2 is questionable, we cannot infer that the argument is sound.

# Doesn't make the argument false necessarily though

- Consider that even if duplicating software is not identical to stealing physical property, it could still cause harm to the property owner
  - because copying the proprietary software program (like the theft of someone's physical property) deprives the property owner of the legitimate use of his or her property
  - or because it is a misuse, misappropriation, or "unfair taking" of another person's property against the property owner's will
- We need to identify in which way software can fit into the existing schemes to protect intellectual property



# Five schemes to protect intellectual property

1. **Copyright** (for music, art, film, literary works and broadcasts)
2. **Patent** (for inventions and processes)
3. **Trademark** (for word, name phrase, or symbol that identify a product or service; a “brand identity”)
4. **Design** (for what a product looks like)
5. **Trade Secret** (for information used in the operation of a business or other enterprise, e.g. a formula, a chemical compound, a blueprint)

# Legislation Relevant in UK

- Copyright and Related Rights Regulations, 2003
- Copyright and Rights in Database Regulations 1997
- Intellectual Property Act 2014
- Data Protection Act 1998
- Computer Misuse Act 1990
- Electronic Commerce (EC Directive) Regulations 2002
- Others:
  - Freedom of Information Act, 2000
  - Disability Discrimination Act, 1995

# Should Software be Free?







# The Case Against Property Rights for Software

- Some argue that while property rights for physical objects make sense, intellectual property rights for software does not.
- most notably, the GNU project (Gnu's Not Unix), has campaigned against copyright protection for software
  - GNU project started in 1984, with the goal to develop an entire Unix-like operating system that was “open” and freely accessible

# GNU and the Free Software Foundation (FSF)

- FSF was formed in 1985 to support of the GNU project
- According to FSF, four “freedoms” are essential for free software, i.e., the freedom to:
  1. run the program, for any purpose;
  2. study how the program works, and adapt it for your needs;
  3. redistribute copies so you can help your neighbour;
  4. improve the program, and release your improvements to the public so that the whole community benefits.

# The Open Source Software movement (OSS)

- OSS, began in 1988, shares many of the FSF goals
  - including the ability of a software user to look at, understand, modify and redistribute the source code for that software.
- Like FSF, OSS requires that the source code for “open source software” is freely available.
- So, both the OSS and FSF movements are similar with respect to their requirements for source code in the software development process.
- There are important differences between OSS and FSF.



# OSS vs FSF

- OSS and FSF have different philosophies:
  - FSF continues to focus on promoting its philosophical position that software should be free.
  - OSS has concentrated its efforts more on promoting the open source model as an alternative methodology to “closed-source” development for software.
- OSS and FSF also differ with respect to requirements for how the software is used “downstream.”

# OSS and FSF (Continued)

- FSF requires that all derivative pieces of software be subject to the original requirements and thus remain “open” and nonproprietary.
  - OSS is more flexible with respect to its derivative software
- FSF requires that users strictly adhere to its GPL (General Programming License) in all derivative uses of its software.
  - OSS supports less restrictive licenses that permit programmers to alter the open source software and to release it as a proprietary product.

# OSS and FSF (Continued)

- generally, OSS is less “anti-commercial” than FSS:
- many in the open source community interact comfortably with members of the business community
  - however, OSS success still poses a significant threat to companies that produce proprietary software

# The Creative Commons

- The Creative Commons (CC) is a nonprofit organisation, launched by Lawrence Lessig and others in 2001.
- Lessig believes that the Internet allows for an “innovation commons”, which needs to be protected just as physical commons (parks, natural resources etc)
- CC aims at providing creative solutions to problems that current copyright laws pose for sharing information.
- CC does not aim to undermine copyright law.
  - Lessig believes that there should be a way to maintain copyrighted works and still make it possible for people to license the use of those works



# CC (Continued)

- Traditional copyright regimes tend to promote an “all or nothing” kind of protection scheme with their “exclusive rights” clauses.
- CC provides a middle ground because it makes possible a “some rights reserved” approach versus an “all rights reserved” policy.
- CC expands the range of creative work available to others legally to build upon and share.
- CC provides a menu of options in its licensing and contract schemes, available for free on its Web site.
- These enable copyright holders to grant some of their rights to the public while retaining others.



# CC provides four options

1. **Attribution**—Permit others to copy, distribute, display, and perform the work and derivative works based upon it only if they give you credit;
2. **Noncommercial**—Permit others to copy, distribute, display, and perform the work and derivative works based upon it only for noncommercial purposes;
3. **Derivative works**—Permit others to copy, distribute, display, and perform only verbatim copies of the work, not derivative works based upon it;
4. **Share alike**—Permit others to distribute derivative works only under a license identical to the license that governs your work.

# CC (Continued)

- CC both encourages the flow of information in digital form and protects the legal rights and interests of artists and authors.
- Artists and authors can be recognised and rewarded, financially and otherwise, for their creative contributions, yet still share their works (or portions of their works) with others.
- This also supports Lessig's notion of an "innovation commons" because it allows authors and artists to build upon the works of others.
- CC also helps to preserve the future of the commons, and it promotes the kind of spirit of cooperation and sharing among creators advocated by FSF and OSS.



# CC for CS

- CC does not recommend the use of their licences to software
  - instead they “strongly encourage” to use either FSF or OSS licences
- BUT:
  - they do say that CC licenses can be used to license databases
  - SW vs DB? ... more on this later!

# GitHub

- “Git” is a free and open source\* (released under the GNU General Public License version 2.0) system designed to handle the version control of projects
  - \*Note: the term “Git” and the logo are however protected
- Git is the core technology of GitHub, which is a software development platform, but also a social platform and user interface



# IP in GitHub

- GitHub has put together a site for developer to be able to use the license they want
  - <https://choosealicense.com/>
- The licence used by the developer WILL BE the licence that remains attached to the code, no matter what happens next, and until the developer decides to change it
- Developers can always make their repository “private”

# “Forking”

**Note:** If you publish your source code in a public repository on GitHub, according to the [Terms of Service](#), other GitHub users have the right to view and fork your repository within the GitHub site. If you have already created a public repository and no longer want users to have access to it, you can make your repository private. When you convert a public repository to a private repository, existing forks or local copies created by other users will still exist. For more information, see "[Making a public repository private.](#)"

- *Forking* is at the core of GitHub spirit: it's the act of creating a personal copy of another user's repository, for it to leave on your account: forked projects retain the same licence as the original (excluding modifications)

# To conclude, as promised:

- **Do students maintain Intellectual Property of the software they produce as part of a university course?**
- *As a taught student you own the rights to the IP that you create as long as it doesn't form part of a larger project being conducted by the University, and uses minimal University resources. If your project falls into this category, you are free to develop the software you have thought of, and publish the code if you wish.  
(communication from the University IP Managers)*