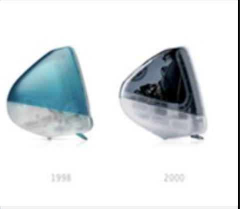
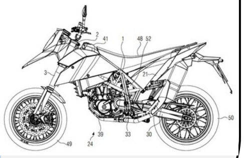
University of Kent

IPR in Software Development

Rogério de Lemos

Intellectual Property

refers to creations of the mind, such as inventions; literary and artistic works; designs; and symbols, names and images used in commerce.

IP rights	Area of application	Legal protection		£
Know-how	In-house info of how things are done. Best known as ***trade secrets***	Confidentiality Agreements		‘Free’
Trademarks	Are signs which distinguish goods and services (***logos***, ***brand*** names)	Registration at patent offices	Google faceb itter adidas Unilever SAMSUNG Coca-Cola MasterCard	£170 (UK)
Industrial designs	Protect the ***appearance*** of items	Registration at patent offices		£50 (UK)
Copyright	***Artistic*** or ***literary*** work	Automatic (© University of Kent 2019)		Free
Patents	Products or processes which have ***functional*** or ***technical*** effect	Registration & examination at patent offices		£5K filing

Agenda

- Software as intellectual property
- IPR in software development
- Protecting software
  - copyright, patent, trade secrets
- IPR and agile development

What Is Intellectual Property?

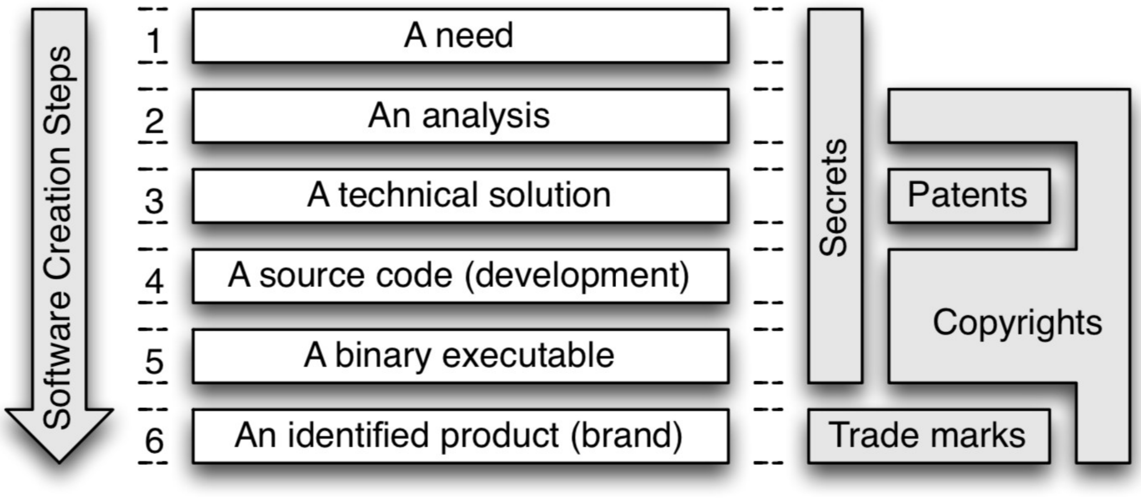
- Intellectual property (IP) is a piece of work that isn't a tangible object
- IP usually comes from creativity and could be could be a manuscript, a formula, a song, or software
- Under the law IP is protected by
  - trade secrets (know-how)
  - trademarks
  - copyright
  - patents
- If someone does steal your property, you have legal grounds to prosecute

Intellectual Property for Software

- Intellectual property for software is computer code or software protected by law under either a copyright, trademark, trade secret, or software patent
- It is important to protect
  - software innovation is valuable to individuals, start-ups, and businesses
  - programmers and businesses treat software as intellectual property, but need to use the law as protection
- When software is treated as intellectual property, one has more control over who gets to use it and how it gets to the public
- Otherwise
  - people might use it without permission
  - lose the chance to get paid when people use the software
  - in extreme cases, one might lose the right to use software **you** created

IPR in Software Development

- Developing software is essentially a creative task
  - relying on the coding and development abilities of a developer
  - transformation of a need into something that is achievable



IPR in Software Development

Types of IPR	Registration required	Software components
Copyright	• No • Protection is automatic • In some countries registration is available and can fulfil essential purposes	• All the creative dimensions embedded in software are protected by copyright, provided they are original • Copyright is the historical and most frequently used means of protecting software • Copyrights protect the code as such, but also the user guides and the graphical elements, such as, icons

IPR in Software Development

Types of IPR	Registration required	Software components
Patent	• Yes, under certain conditions	• Patents are meant to protect the functional dimension of software, by providing potential ownership of new and inventive technical effects implemented by the program

IPR in Software Development

Types of IPR	Registration required	Software components
Trade marks	• It is highly advisable to seek registration	• Protects an essential aspect of software, be it of a visual or textual nature (via either a logo or a word). A trade mark is an essential protection in order to differentiate assets on a given market

IPR in Software Development

Types of IPR	Registration required	Software components
Confidential Information (trade secrets)	• No	• Specific and identified information can be protected through contractual arrangements

IPR in Software Development

Types of IPR	Registration required	Software components
Industrial Design	• Registration is generally recommended, even though unregistered designs can be protected	• Protects the graphical user interfaces under certain requirements

Protecting Software through Copyrights

- Copyright
  - original works of authorship fixed in any tangible medium of expression
- Copyright protects (internationally)
  - against word-for-word copying
- Holding copyright to software code, allows
  - **Make copies** of your software code
  - **Sell or give away** the code
  - **Make a "derivative work,"** which is a second software that uses a lot of the original code
  - **Post the code** somewhere, or otherwise display it

upcounsel - Intellectual Property Protection for Software: What to Know

Protecting Software through Copyrights

- Stealing from software is more complicated than just plagiarising, because software is more than just code
- Software is an invention, or an idea
  - copyright law only protects how that idea is written down
  - software program does something specific, protecting against copying might not be enough
  - someone could use different code but still steal your invention

upcounsel - Intellectual Property Protection for Software:
What to Know

Protecting Software With a Patent

- Patent is needed to protect a process
  - systems
  - functions
  - solutions to computer problems
- Unlike copyright law, patent law protects the invention itself
  - someone can't create a software program with different code that does the exact same thing your software does
  - patent doesn't protect your specific lines of code against plagiarism the way copyright does
- If information is included in the published patent application, that information is no longer a trade secret

upcounsel - Intellectual Property Protection for Software: What to Know

Difficulties With Patenting Software

- The difficulties around software patents stem from the definition of what is patentable
- It is not patentable
  - Abstract ideas
  - Natural phenomena
  - Laws of nature
- Key criteria for patenting software
  - Your software has to be **useful** or have a specific application
  - It has to be **new**
  - It has to have an **"inventive step"** that is not obvious to people in the field.

upcounsel - Intellectual Property Protection for Software: What to Know

Difficulties With Patenting Software

- Algorithms cannot be patented because that's an abstract idea
  - if the algorithm has a specific purpose and application in the software, then the software may qualify for a patent
  - if your software solves a specific technology problem in a way you can discuss in detail, then you could qualify for a patent
- Abstract ideas are where most software patent applications run into trouble
  - proving that software is not an abstract idea, but qualifies as an invention, is difficult

upcounsel - Intellectual Property Protection for Software: What to Know

Difficulties With Patenting Software

- Examples of successful software patents:
  - Amazon "One-Click-Buy" software, U.S. Patent No. 5960411
  - mp3 audio software, U.S. Patent No. 5579430

upcounsel - Intellectual Property Protection for Software: What to Know

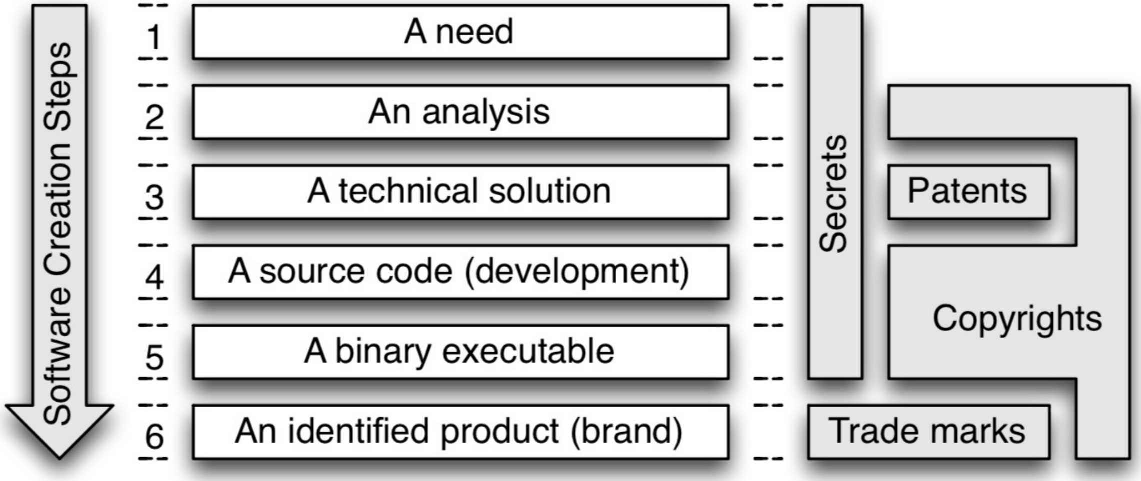
Protecting Software as a Trade Secret

- A **trade secret** is information you or your company has that other people don't have
  - provides an advantage over competition
- We don’t apply for a trade secret
  - take "reasonable measures" to keep the software a secret
    - Keep the software away from the public
    - Have employees sign non-disclosure agreements
    - Do exit interviews with employees who are leaving to make sure they aren't bringing IP with them
    - As soon as an employee quits or is fired, take away all their file and data access
    - Investigate any suspicious employee activity
    - Keep IP data stored in compartments, and only give access to employees who need it

upcounsel - Intellectual Property Protection for Software: What to Know

What about Agile Development?

- Do you remember this diagram?



- The people working on each step don't usually communicate
- Clients and customers don't see the software until the developers are finished

What about Agile Development?

- Agile software development
  - developers work together on different parts of the project
    - developed in burst, e.g., sprints
  - communicate as the software evolves
  - solve problems as they appear
  - clients or customers also see the partly finished software

- Different from traditional software development
  - who worked on which piece of code?
    - more difficult to figure out who owns IP!
  - pieces of software might only be parts of code and might not function on their own without the rest of the software

upcounsel - Intellectual Property Protection for Software: What to Know

IP Problems in Agile Development

- Identifying IP software is important to copyright, trade secret, patent, and trademark
- Common issues add to the difficulties in identifying IP during agile development
  - A short development cycle
  - Less time to think about or document possible IP
  - Lots of people working on pieces of the same code
  - Difficulty seeing how small innovations during different work cycles fit together to create IP
  - Less communication between the developers and company lawyers who could advise about IP
    - DevOps a solution?

upcounsel - Intellectual Property Protection for Software: What to Know

IP Solutions and Techniques in Agile Development

- People using agile development have to adjust the techniques they use for finding and detailing IP
    - Have employees and contractors sign nondisclosure agreements?
    - Make sure employee and freelancer contracts include clauses about who any IP produced belongs to
    - Create security measures around every project, like document management and data backups
    - Encourage employees to be inventive by recognizing their innovation with awards or bonuses
    - Mark in-development projects with copyright notices to add extra protection
    - Treat projects like trade secrets by adding trade secret protocols around developing projects
    - Store in-development projects away from non-secret projects
    - Teach employees about trade secret practices

Client Feedback: A Third-Party Agile Development Solution

- Client or customer gives feedback on the unfinished software
    - protecting IP is difficult!
- Before starting agile development
    - include a nondisclosure agreement or a product advisor agreement
        - "feedback clause" within these agreements should state that any ideas or feedback that the client produces is free for the developer to use

upcounsel - Intellectual Property Protection for Software: What to Know

Filing for Patents during Agile Development

- Start filing provisional patent applications during agile development
- Provisional patent application gives you a filing date for your patent, but it isn't your full application
  - it gives you a year to create the non-provisional application
  - provisional patent applications are a way to protect software that is in development

upcounsel - Intellectual Property Protection for Software: What to Know

Summary

- In term of protecting IP, software is different
- There is combination of ways
  - copyright
  - patent
  - trade secrets
- Agile development makes the things more complicated
  - users or customers are involved from the beginning
  - no clear boundaries between developers, and what is already developed