

# Legal and Business Aspects of Software Engineering

Software is developed in a complex **legal** and **economic** framework.

Every software developer needs to be **aware** of parts of the framework.

**Law usually develops incrementally.**

Changes in laws usually follow changes in technical world.

Lawyers and politicians typically have poor technical backgrounds.

The interpretation of many laws is unclear as they have may never have been tested in court because of the cost of litigation.

Strange analogies are often made between new technological paradigms and old world systems.

# Legal topics in Software Development

- **Jurisdiction** (international, federal, state laws)
- **Intellectual property** (copyright, patent, trade secrets)
- **Contracts and licenses**
- **Privacy**
- **Free speech** and its limitations (government secrets, obscenity)
- **Employment** (personnel, your next job, etc.)

# Copyright

**Copyright** is Federal law, which applies to “literary works”.

Originally applied to textual materials, but gradually extended to cover text, music, photographs, designs, **software**, ...

Copyright applies to the **expression of ideas** (e.g., the words), not to the ideas themselves, nor to physical items.

## **Software**

Copyright applies to the **program instructions**, but not to the concepts behind the instructions, nor to the files on disk or printouts of the programs instructions.

- If you are employed, copyright in the code that you write belongs to **your employer** (e.g., if you work for a Cornell department).
- If you work free lance, you must **have a contract** with the organization you are working for that is explicit about who owns the copyright in work that you do.
- If you do not own the copyright, you need permission to copy, or use the software, **even if you wrote the code** (e.g., you cannot make a copy for your personal use).

If your project is for a start-up company or other entrepreneurial venture, it is essential to agree on ownership of software and sharing of potential rewards as part of the **feasibility study**.

If in any doubt, ask for advice.

**Example:**

Film **The Social Network**

### When software is derived from other software:

- Copyright in the **new code only** is owned by new developer or employer
- Conditions that apply to **old code** apply to derived work

If you write S, which includes code derived from A and B, you cannot copy, use, distribute or license S unless the copyright owners of both A and B give permission.

When creating a software product, you should have **documented** rights to use everything from which it is derived.

When software is written, who owns the copyright?

How can somebody else be permitted to use the software?

How can copyright be transferred to somebody else?



When software is written, who owns the copyright?

The person who writes the software

Except works for hire, where the employer owns copyright

How can somebody else be permitted to use the software?

By permission from the copyright owner  
(usually a license)

How can copyright be transferred to somebody else?

Copyright is intellectual property that can be sold or given away  
(usually a contract)

You are employed for company X writing software.

When you leave, who owns your work?

What use can you make of the work?

You are employed for company X writing software.

When you leave, who owns your work?

The company (work for hire)

What use can you make of the work?

None, without permission of the copyright owner.  
(Perhaps some minor use under "fair use".)

You do free-lance work for company X.

When you finish, who owns your work?

What use can you make of the work?

You do free-lance work for company X.

When you finish, who owns your work?

It depends on the circumstances.

Have a written contract before beginning work.

What use can you make of the work?

If you hold the copyright -- unrestricted (unless you have signed away some of your rights).

Otherwise -- none without agreement  
(perhaps some minor use under "fair use").

# Software Contracts

## Contracts allow software to be licensed or sold

- A contract is an agreement in exchange for some consideration (e.g., money)
- It should be a written document signed by both parties
- A contract is enforceable by courts

For simple agreements, an exchange of letters may be a convenient form of contract.

A typical contract might include wording that defines:

- The software that is covered by the agreement
- Who owns the software
- Whether any patents or other intellectual property are involved and who owns them
- Payment amounts and dates
- Whether the agreement is permanent or temporary
- Whether the agreement is exclusive or non-exclusive
- Terms about contingencies, termination, problems, and difficulties
- Any other terms and conditions that are agreed by both parties

# Open Source Software

Open source software is an important part of modern computing that does not fit well into contract law.

## Examples of open source licenses

### BSD license:

A class of extremely simple and very liberal licenses for computer software, originally developed at the University of California at Berkeley (UCB). Often suitable for academic projects.

### Apache license:

A permissive license that does not require a derivative work of the software, or modifications to the original, to be distributed using the same license. Widely used, e.g., Android.

### GNU public license:

A less permissive license, which requires that derived works can only be distributed under the same license terms.



# Patents

## Patents apply to inventions

- Should be: non-obvious, novel, useful
- Requires a complex process of patent application
- 17 years from award (20 years from application)

Copyright applies to the expression of ideas, patents to the ideas themselves.

## Problems with software patents

- Usually difficult to know where ideas originate.
- Poor quality of patent examiners has lead to broad patents for routine computing concepts.
- There are often hundreds of patents covering essentially the same idea.
- International differences.

The Supreme Court has recently created precedents that may lead to stricter examination of patent applications

New law on first to file may make the software situation even worse

# Patent Abuse

Because of the huge number of dubious patents that have been granted, companies can often use the threat of patent litigation as a commercial weapon:

- **Large companies** harass smaller competitors to delay the introduction of competing products and to force them into expensive litigation.
- **Patent trolls** are companies whose entire business is to threaten patent litigation. They buy large numbers of patents and extort money from companies whose products use ideas related to the patents.

# Trade secret and Non Disclosure Agreements

**Trade secret:** confidential business information

## Examples

- Specification of a product before it is publicly announced
- Source code of a commercial product

## Legal definition

"... information, including a formula, pattern, compilation, program, device, method, technique, or process that derives independent economic value from not being generally known and not being readily ascertainable and is subject to reasonable efforts to maintain secrecy."

*Uniform Trade Secrets Act*

## Non-Disclosure Agreement

Legal agreement not to disclose trade secrets. It is often reasonable to sign a non-disclosure agreement, but **read it carefully before signing.**

# Trade Secret

## Trade secrets

- Owner must make **reasonable efforts** to maintain secrecy, e.g., label information “Confidential” and restrict access to designated people.
- A trade secret does not expire, as long as reasonable efforts are made to keep it secret.
- If somebody leaks a trade secret without authorization, it remains a trade secret.
- Competitors may not use secrets obtained through extraordinary means.
- If you learn trade secrets when working for one employer, you must not disclose them to another employer.

## Example

If you work for some companies, e.g., Apple, you may be required to sign a non-disclosure agreement that prevents you telling anybody even what project you are working on.

# Privacy

Laws and social norms about privacy are changing rapidly. In general the following are **invasions of privacy**:

- intrusion into the private life of somebody
- appropriation of name or likeness
- unreasonable publicity
- false light

The borderline between appropriate **data mining** and illegal invasion of privacy is vague.

Be very careful about collecting personal data without the knowledge of the individual, e.g., medical or financial data.

There are special laws about privacy of **children**.

**European law** is stricter in protecting privacy than US law, e.g., recent Google policy on recording Wi-Fi locations.

# Privacy in the workplace

Test for employers/employees - "Do you have a reasonable expectation of privacy?"

Work-related material on business machines is definitely not private.

Some organizations, e.g., most universities, treat private email on business machines as private, but this is not the law.

# Privacy in Email

Test for employers/employees - "Do you have a reasonable expectation of privacy?"

Work-related material on business machines is definitely not private.

Some organizations, e.g., most universities, treat private email on business machines as private, but this is not the law.



# Business Email

The Electronic Communications Privacy Act (1986) says all business communication belongs to that business.

Deleting email can be ruled intentionally destroying company records.

Email is frequently subpoenaed in legal disputes:

Never send anything by email that you would not be prepared to be seen as evidence in a court of law.

# Private information

## Private information

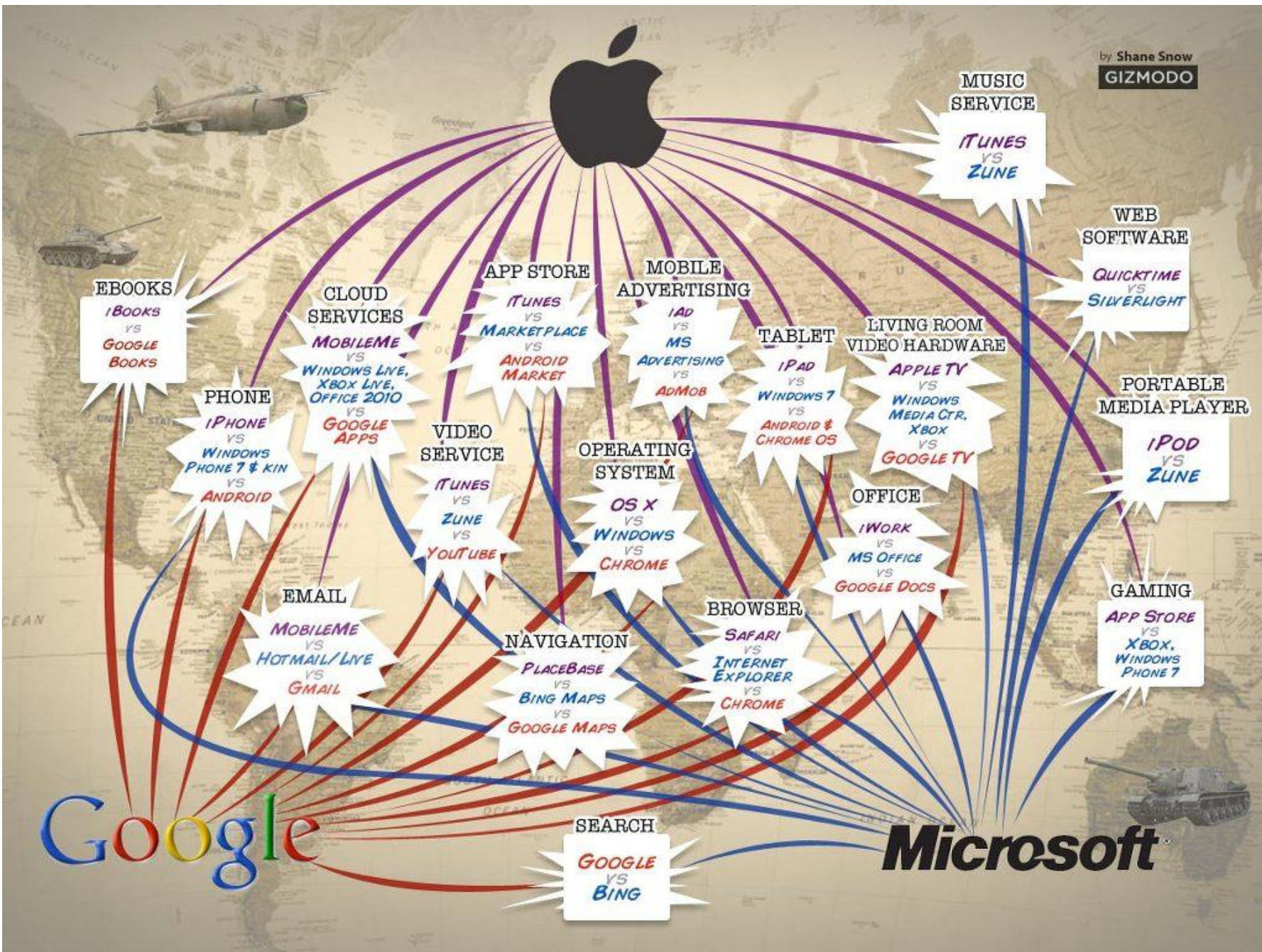
As a software developer, and particularly if you work as a system administrator, you may come across private information, e.g.,

- Personal records, such as medical records, financial information, etc.
- Information about dishonest, immoral, or illegal activities
- Corporate secrets

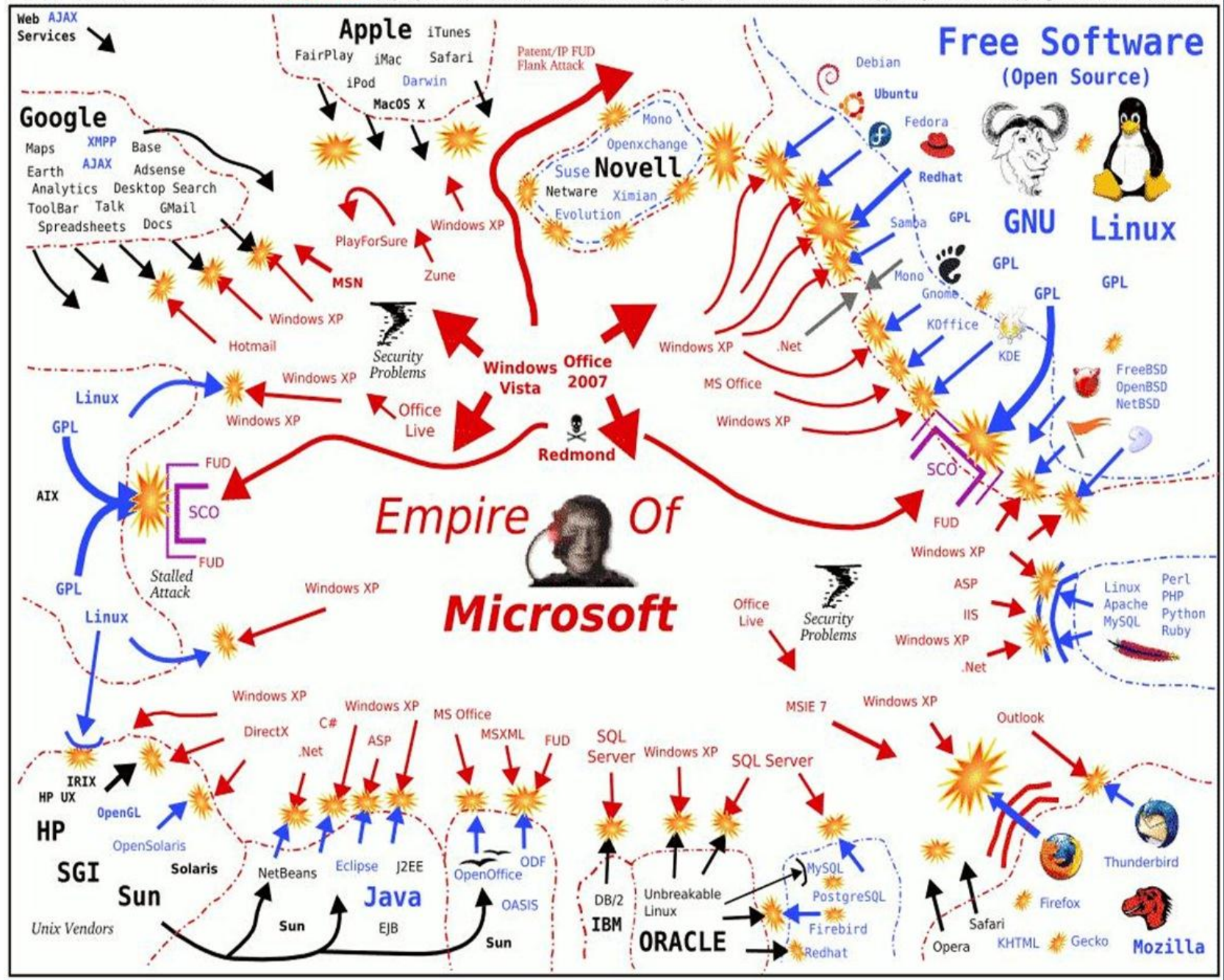
General rule: **keep the information private.**

If in doubt, e.g., the information implies serious criminal activities, consult your supervisor or a senior manager.

**Do not take action yourself.**







# Sources

- Cornell University, CS 5150, William Arms
- Lawful Software Engineering, German, Weber, Di Penta
- <http://blog.p2pfoundation.net/why-apache-defeated-the-gpl-license-developer-freedom-vs-user-freedom/2013/01/21>