

Legal perspectives on system development

Liability, Litigation risk, 'Professional' standards, and Ethics

<http://cyberlawcentre.org/seng4921/cse.htm>

David Vaile
Executive Director
Cyberspace Law and Policy Centre
Faculty of Law, University of NSW
<http://www.cyberlawcentre.org/>

Outline

Strange bedfellows

Legal system

Liability

Software development – immature?

Examples:

- Consumer protection
- Product liability
- Professional liability
- Anti-trust
- Copyright
- Software patents
- Privacy
- Spam

My background

Tech support manager: JAM Software

- OS, expert system, utils - UCD

Privacy Commissioner's Office: policy

- Big government projects

Public interest law: PIAC, RLC, Legal Aid

- Test cases, Consumer, Crime

Virtual community SysAdmin: MSP

Web exec. producer: Access Online

Australian Privacy Foundation

Software, Law and Ethics

Strange bedfellows

How the law is made, and works

Differing Principles and standards

Risks in software development

Examples:

- Consumer protection
- Product liability
- Professional liability
- Anti-trust: abuse of monopoly
- Intellectual property: copyright, patents
- Privacy
- Spam

Features of the legal system

Main divide: Criminal <-> the rest

Criminal

- Launched by state, trial, conviction or acquittal

Civil

- Sued by other party, damages, restitution

Sources

- **Statutes** ("Laws") set rules, **Cases** interpret them
- **Jurisdiction**: which laws and courts
- **Appeals** to higher court
- **Precedent** is critical in cases

Obligations: from Statutes and Contracts

Everything is arguable (if you lose, \$\$ costs)

What shapes the law?

Ongoing struggle between interests

Commercial reality

Technical reality

Public standards

International affects (indirect)

Different standards

Liability

- Is it against the law?

Litigation risk

- Will you be caught, sued or prosecuted?

'Professional' standards

- Will your peers reject you?

Ethics

- Will your children & friends reject you?

Development risk factors

20% coding and engineering – ignore?

80% analysis, communication, revision

User-Centred Design & Risk Management

Neglected but critical

Early vs. late error discovery

'User sovereignty'

Hypothetical

Most software projects fail

- \$, time, scope, quality (for User)

Many break various standards, but...

You could do it accidentally...

Or be asked/tempted to deliberately

Your own position

Your employers

The 'victim's position'

What matters?

Breaking the law?

Liability

Getting caught?

Enforcement

Losing your job?

Professional

Losing your reputation?

Ethics

Or just building crap?

Self respect

Consumer Protection

Based on consumer/vendor relation

Assumes imbalance

Statutory Warranties – fit purpose

Contractual waiver?

Misleading and deceptive conduct

Unfair Contracts

Can be Strict Liability – State Bank

Tort/ Negligence

Product liability

Duty of Care, special relationship

Act or omission

Causation

Foreseeability of harm

Proximity

Consumer protection hypothetical

See hypothetical [example](#)

Professional Liability

Nature of Profession?
Membership of Professional body
Registration required to work?
Self-regulation
Insurance
Peer attitudes
Reputation

Anti-trust: Abuse of Monopoly

Competition policy
Monopoly
Example: *MS v DoJ re Netscape*
Political involvement
Practical significance

Anti-trust hypothetical

See hypothetical [example](#)

Intellectual Property

Purpose:

Copyright Act: form, not substance

- No registration
- Digital Agenda

Patents Act: the idea, not the form

Circuit Designs

Free Trade Agreement

Copyright

Copyright Act:

- Exclusive right to control exploitation

No registration

Actual text, code or implementation

Licences with conditions and fees

Technological Protection

- 'Digital Rights Management' tools
- DMCA and contracting away user rights

Copyright and Public Domain

Differences in Australia, US...

Fierce battle: maximalist v PD?

'Public Domain'

Open Source software: GPL, copyleft

Open Content

- Creative Commons – US, global?
- Free for Education – Australian

Business models

Patents and software

Right to deny access

Requires registration

Expensive to fight

Patentable material?

E-business patents

- Amazon 1-Click web shopping cart

Gene sequence patents

- Bioinformatics – human genome race

Current patent battles

Resistance to patentability of software
EU Commission recommends, Parl. Rejects
CSIRO v. US computer industry – wireless
Linux?

Why are software patents a danger?

- Locking up pure ideas? Mathematics? Stallman
- Not just open source
- Impossible to ascertain if infringing
- Patent Offices too lax and inexperienced? \$\$ motive
- Very expensive
- Only works if you have a huge portfolio

Privacy

Right to be left alone
Defeat of Australia Card, Privacy Act
Limited Rights of data subjects
Restricts what technology can do
Requires security
Affects everyone

Privacy Hypothetical

See hypothetical [example](#)

Spam

Spam Acts: Australia, USA, California
Unsolicited commercial electronic
message
Single message
Address harvesting
Penalties
Surveillance
Workplace privacy bill NSW

Spam hypothetical

See hypothetical [example](#)

Questions?

Conclusion

David Vaile
Executive Director
Cyberspace Law and Policy Centre
Faculty of Law, University of NSW
<http://www.cyberlawcentre.org/>