Social and Legal Issues in Informatics

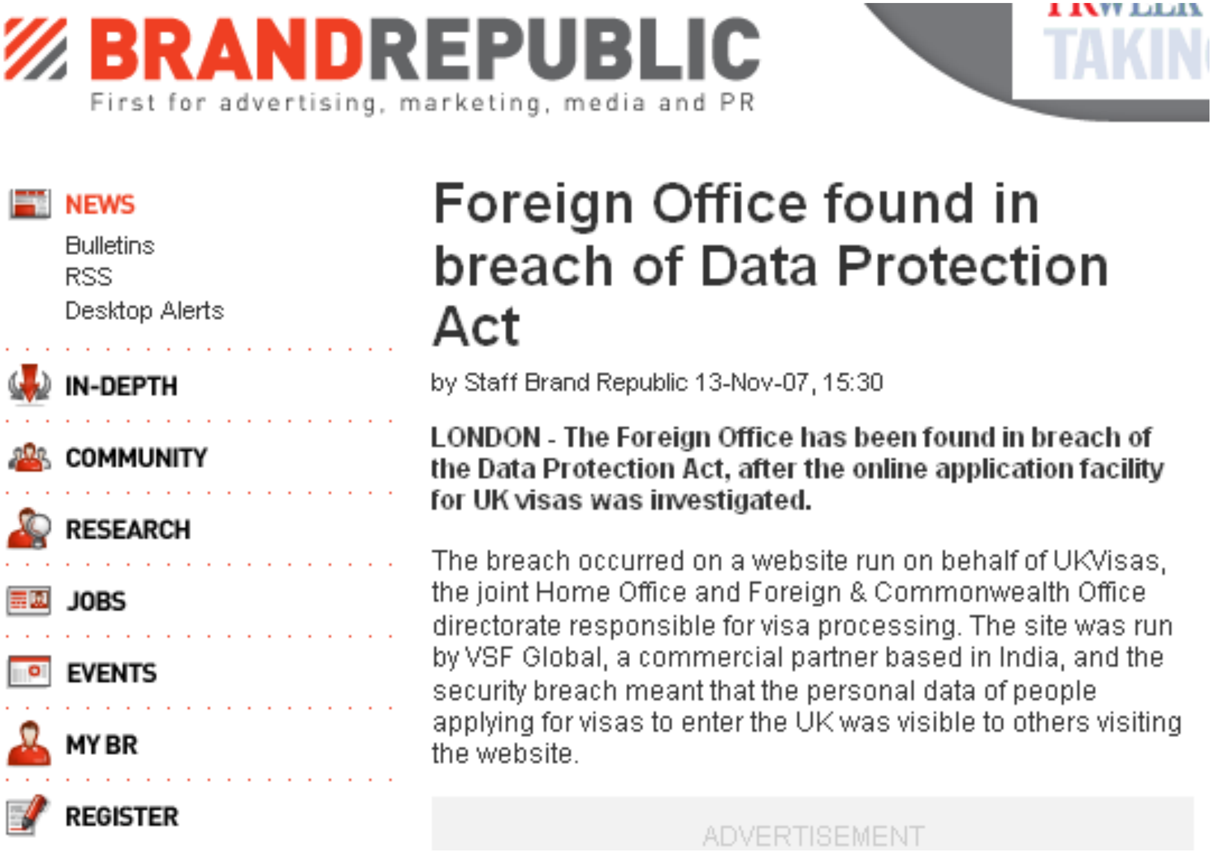
MSc Management – IS and Services Science

Contracts and Liabilities

Giovanna Di Marzo Serugendo
Giovanna.Dimarzo@unige.ch, room B 235, 022 379 00 72

(Some slides provided by Roger Johnson, Birkbeck College, University of London, UK)

Foreign Office Breaches Data Protection Act



Foreign Office found in breach of Data Protection Act

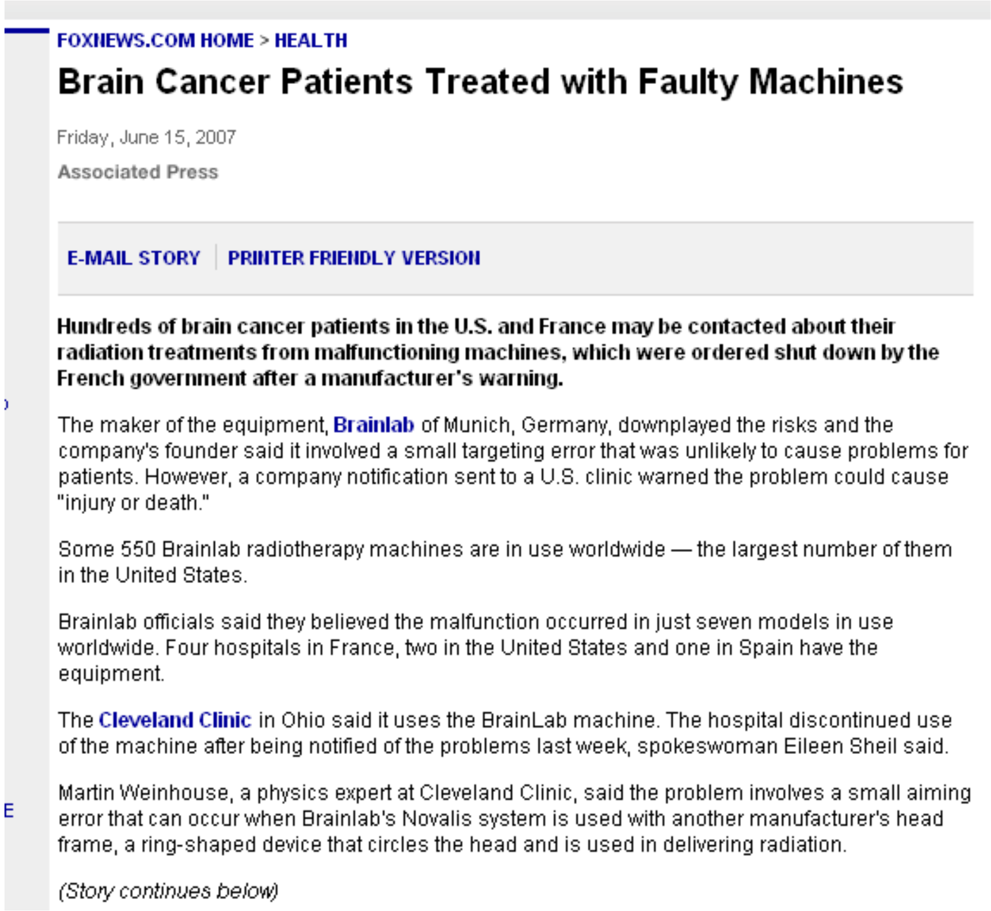
by Staff Brand Republic 13-Nov-07, 15:30

LONDON - The Foreign Office has been found in breach of the Data Protection Act, after the online application facility for UK visas was investigated.

The breach occurred on a website run on behalf of UKVisas, the joint Home Office and Foreign & Commonwealth Office directorate responsible for visa processing. The site was run by VSF Global, a commercial partner based in India, and the security breach meant that the personal data of people applying for visas to enter the UK was visible to others visiting the website.

http://www.brandrepublic.com/News/766695/Foreign-Office-found-breach-Data-Protection-Act/
http://www.computerworld.com.au/index.php?id=1181601580&rid=-255

Brain Cancer Machines Faulty



FOXNEWS.COM HOME > HEALTH

Brain Cancer Patients Treated with Faulty Machines

Friday, June 15, 2007

Associated Press

E-MAIL STORY | PRINTER FRIENDLY VERSION

Hundreds of brain cancer patients in the U.S. and France may be contacted about their radiation treatments from malfunctioning machines, which were ordered shut down by the French government after a manufacturer's warning.

The maker of the equipment, Brainlab of Munich, Germany, downplayed the risks and the company's founder said it involved a small targeting error that was unlikely to cause problems for patients. However, a company notification sent to a U.S. clinic warned the problem could cause "injury or death."

Some 550 Brainlab radiotherapy machines are in use worldwide — the largest number of them in the United States.

Brainlab officials said they believed the malfunction occurred in just seven models in use worldwide. Four hospitals in France, two in the United States and one in Spain have the equipment.

The Cleveland Clinic in Ohio said it uses the BrainLab machine. The hospital discontinued use of the machine after being notified of the problems last week, spokeswoman Eileen Sheil said.

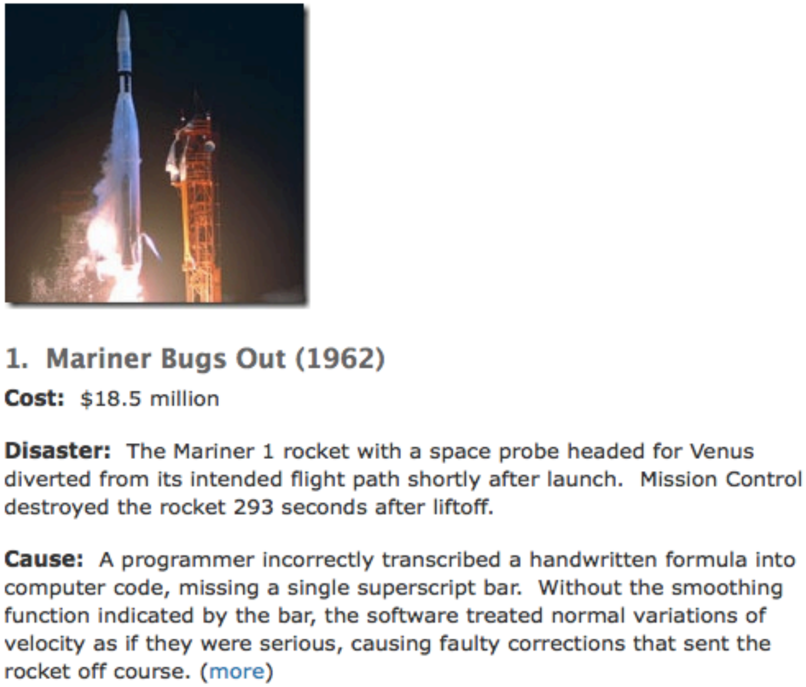
Martin Weinhouse, a physics expert at Cleveland Clinic, said the problem involves a small aiming error that can occur when Brainlab's Novalis system is used with another manufacturer's head frame, a ring-shaped device that circles the head and is used in delivering radiation.

(Story continues below)

http://www.foxnews.com/story/0,2933,282994,00.html

http://annual-report.asn.fr/PDF/cancer-radiotherapy.pdf

Famous Software Disasters

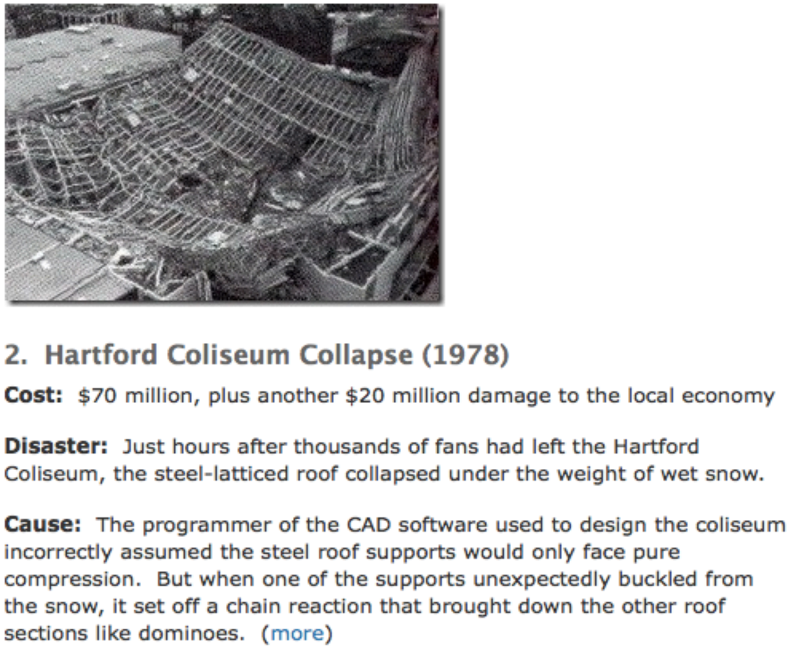


1. Mariner Bugs Out (1962)

Cost: $18.5 million

Disaster: The Mariner 1 rocket with a space probe headed for Venus diverted from its intended flight path shortly after launch. Mission Control destroyed the rocket 293 seconds after liftoff.

Cause: A programmer incorrectly transcribed a handwritten formula into computer code, missing a single superscript bar. Without the smoothing function indicated by the bar, the software treated normal variations of velocity as if they were serious, causing faulty corrections that sent the rocket off course. (more)



2. Hartford Coliseum Collapse (1978)

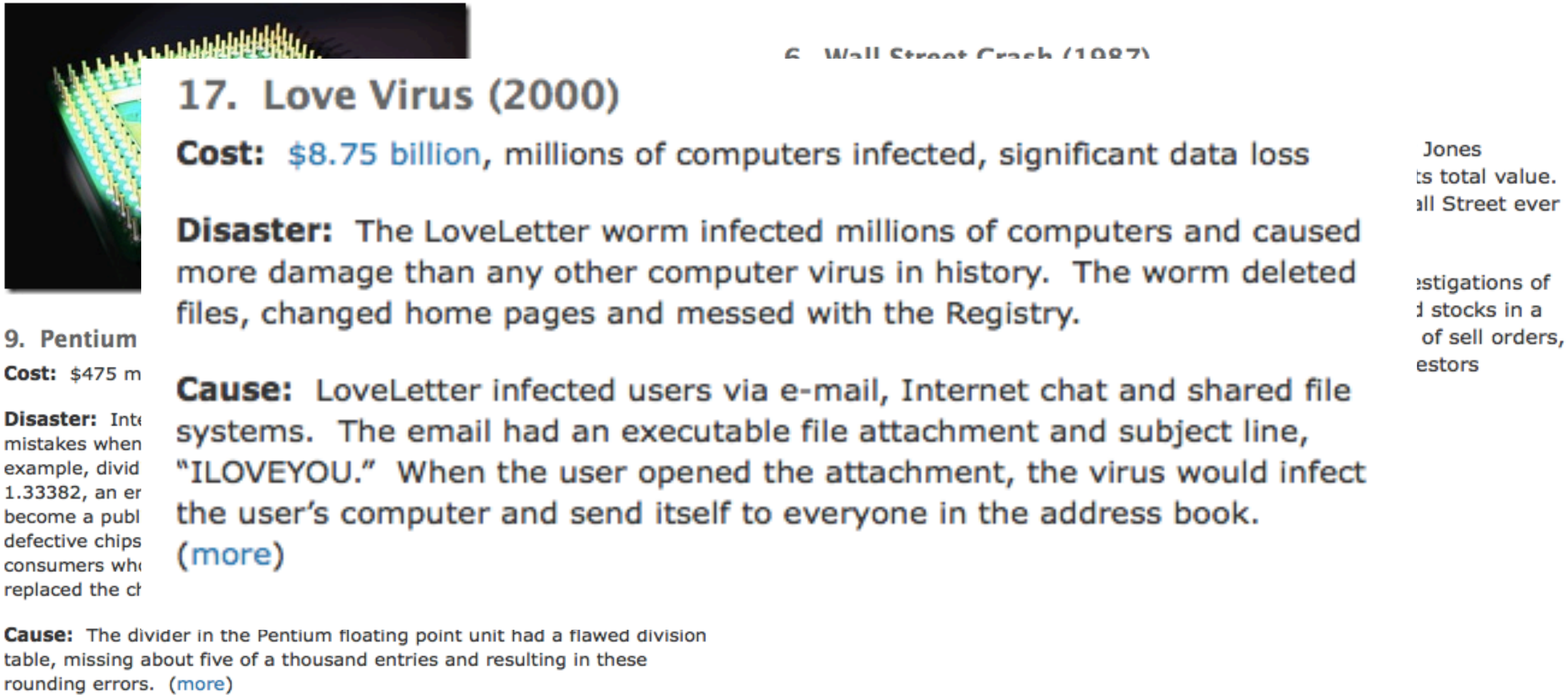
Cost: $70 million, plus another $20 million damage to the local economy

Disaster: Just hours after thousands of fans had left the Hartford Coliseum, the steel-latticed roof collapsed under the weight of wet snow.

Cause: The programmer of the CAD software used to design the coliseum incorrectly assumed the steel roof supports would only face pure compression. But when one of the supports unexpectedly buckled from the snow, it set off a chain reaction that brought down the other roof sections like dominoes. (more)

http://www.devtopics.com/20-famous-software-disasters/ (part 1)
http://www.devtopics.com/20-famous-software-disasters-part-2/ (part 2)
http://www.devtopics.com/20-famous-software-disasters-part-3/ (part 3)
http://www.devtopics.com/20-famous-software-disasters-part-4/ (part 4)

Famous Software Disasters



17. Love Virus (2000)

Cost: $8.75 billion, millions of computers infected, significant data loss

Disaster: The LoveLetter worm infected millions of computers and caused more damage than any other computer virus in history. The worm deleted files, changed home pages and messed with the Registry.

Cause: LoveLetter infected users via e-mail, Internet chat and shared file systems. The email had an executable file attachment and subject line, "ILOVEYOU." When the user opened the attachment, the virus would infect the user's computer and send itself to everyone in the address book. (more)

9. Pentium

Cost: $475 m

Disaster: Inte
mistakes when
example, divid
1.33382, an er
become a publ
defective chips
consumers wh
replaced the ch

Cause: The divider in the Pentium floating point unit had a flawed division table, missing about five of a thousand entries and resulting in these rounding errors. (more)

6. Wall Street Crash (1987)

Jones
ts total value.
all Street ever

estigations of
d stocks in a
of sell orders,
estors

http://www.devtopics.com/20-famous-software-disasters/ (part 1)
http://www.devtopics.com/20-famous-software-disasters-part-2/ (part 2)
http://www.devtopics.com/20-famous-software-disasters-part-3/ (part 3)
http://www.devtopics.com/20-famous-software-disasters-part-4/ (part 4)

Overview

- Some Questions (with not always an answer …)
- Contracts: Liability clauses
- The particular case of software
- Examples
- Acts (UCTA, CPA)
- Types of Software and Outcomes
- Good practice in developing software
- A Last Example (on-going …)
- Software Contracts

Some interesting questions

- Should software companies be liable for software failures?
- What is the definition of negligence with respect to software development?
- Do existing laws account for the unique characteristics of software engineering?
- What ethical responsibilities do software engineers have to users?
- How should the terms “appropriate use” and “appropriate care” be defined in software liability law?
- What influence have corporations had in the development of existing law?
- Is software a tangible product? Tangibility is an important concept in products liability law.
- What is the concept of information liability? Should software companies be liable for information generated by their software?

http://cse.stanford.edu/class/cs201/projects-95-96/liability-law/

Some interesting questions

- Would increased liability stifle the quick release of new software?
- What would be the economic ramifications of an increased level of liability? Would such a change discourage the development of software for medical and other high risk fields?
- Is a computer program a product or a service?
- If an expert system using artificial-intelligence gives bad advice, should the programmers be held liable?
- Should programmers be considered professionals and thus subject to malpractice suits?
- What risks should users naturally assume when using software?
- Because computer programming is extremely complex, should the doctrine of strict liability apply to programmers in order to induce them to write bug-free software? Is such software possible?

http://cse.stanford.edu/class/cs201/projects-95-96/liability-law/

… and also

Who is responsible?

- Programmer
- Software Project Designer/Architect
- Consultant
- “Old” programmers (no longer in the same project)
- CEO of software delivering company
- Client (not able to specify what it wants correctly)
- Software uses another piece of software
- What about Open Source Software?

Contracts

Suppliers insert clauses in contract:

- To limit their liability in case software is defective
  - Limit to the purchase price / maximum value

Law:

- Limits the effects of such clauses
- Unfair Contract Term Acts 1977
  - It is not possible to limit the damages payable if a defect in a product causes death or personal injury
    - E.g. “The McDonald Coffee Case”
    - http://www.lectlaw.com/files/cur78.htm

The Case of Software

Sales of Goods Act 1979

- "Goods sold must be fit for the purpose for which such goods are commonly supplied"

- … if a consumer buys a software that does not work as expected he should be refunded

- BUT
  - Is a "Software a Good?"

The Case of Software

Two cases

- Software comes with “shrink-wrapped license”
  - Software bought in a “box”, license is on the back of the box
  - Buyer is a private individual
  - Considered as a Good
    - Sales of Goods Act 1979
- Bespoke Software
  - Not considered as a Good
    - Supply of Goods and Services Act 1982
    - “Reasonable skill and care” sufficient to protect supplier

The Case of Software

But in both cases

- Unfair Contract Term Acts 1977
  - Liability is limited or excluded only to some reasonable extent
    - Liability is no longer limited as specified in the contract
    - The limits have to be reconsidered reasonably
      - Depends on case and on Court …

St Albans Example

St Albans City and District Council vs International Computer Ltd (ICL) - 1988

Case

- Council ordered ICL to provide a computer system for computing local taxation
- ICL used its standard terms in the contract
  - "liability will not exceed the price or charge payable for the item of Equipment, Program or Service in respect of which liability arises or £100'000 (whichever is the lesser)"
- Errors in software + incorrect advice from ICL manager
  - Residents were undercharged, Council lost £1.3Million

St Albans Case

Judge decisions

- 1. Software was not fit for purpose
- 2. ICL manager has been negligent
  - ICL was in breach of contract
  - Clause of limiting liability had to be measured against reasonableness
    - ICL had liability insurance of £50 Millions
    - Council was not usual business consumer
      - Could not have its own insurance against commercial risks
    - £100’000 was not reasonable
  - ICL to pay £1.3 Million (later reduced by £484’000)

General Motors vs Johnston

General Motors vs Johnston (Lewis) – 1987

Case

- Lewis was driving a GM car bought two days earlier
- Car stalled in the middle of intersection
- Lewis effort to restart car failed
- A tractor engine collided with car – injury + death
- Cause:
  - an electronic control module controlled fuel delivery
  - PROM relayed command to the engine
  - PROM was defective

http://www.badsoftware.com/johnston.htm

General Motors vs Johntson

Judge Decisions

- GM apparently new about the problem with PROM
- GM had a new version
- Experts assured that the car was actually not running and PROM caused problem
  - GM to pay compensatory damages (> $7.5Millions)

http://www.badsoftware.com/johnston.htm

Overview

Defective Software

- Common law
  - Need to establish duty of care
    - “legal requirement that a person exercise a reasonable standard of care to prevent injury of others “
  - Need to establish breach of duty of care
  - On the consumer side
- Breach of Contract
  - Unfair Term Contract Act
  - Regulations on Unfair Terms in Consumer Contract
- Consumer Protection Act
  - Product Liability
  - Shift from consumer to producer
  - Producer needs to provide proofs for its defence
  - Consumer does not need to prove fault on the producer part
  - Question
    - Is “software a product?”
      - No: not movable, it is information
      - Yes: similar to electricity, information is treated as product in our society
    - Software development is a service
  - EC: “software should be categorised as a product”

http://www.journalonline.co.uk/article/1000702.aspx

UCTA

The Unfair Contract Terms Act

- Places restrictions on the contract terms businesses can agree to
- Define rules for the ways in which vendor businesses can use exclusion clauses to limit liability in certain areas
  - excluding liability for death or injury is not permitted in any circumstances
  - excluding liability for losses caused by negligence is permitted only if it is reasonable
  - excluding liability for defective or poor-quality goods is also permitted only if it is reasonable

Test of “reasonableness”

- Not defined precisely, but courts usually take into account:
  - information available to both parties when the contract was set up
  - negotiated or standard form contract
  - whether the purchaser had power to negotiate better terms
- Businesses don't have the same protection as individual consumers.
  - Consumer contract excluding liability for defective goods is automatically invalid.
  - Business client must check terms in advance

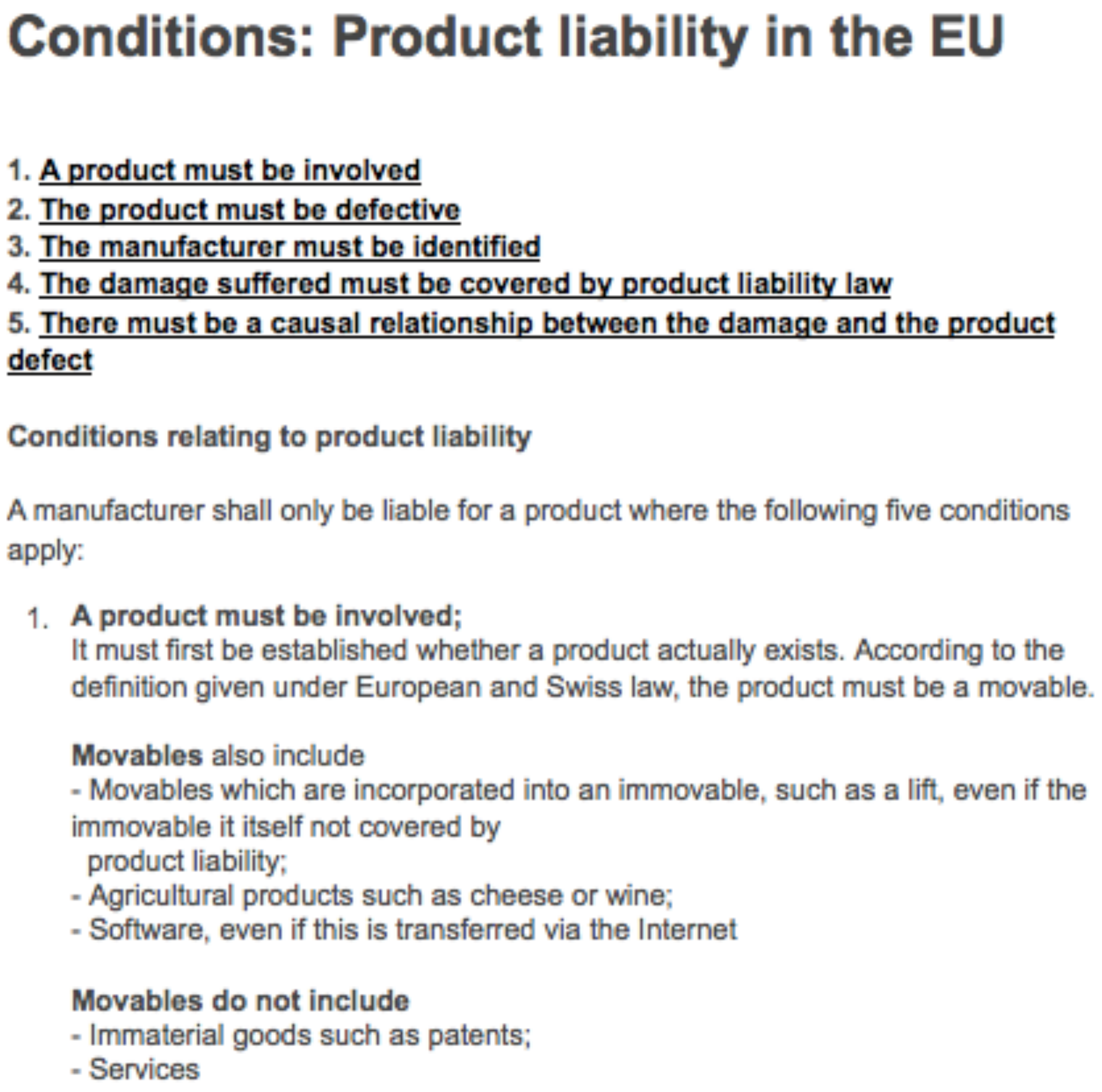
Department of Trade and Industry
http://www.dti.gov.uk/consumers/buying-selling/sale-supply/page8599.html

Consumer Protection Act 1987

Consumer Safety and Product Liability

Consumer Protection Act 1987

- UK law for EU Directive 85/374/EEC
- Imposes strict liability on producers for harm caused by defective products
- People injured by defective products do not need to prove that:
  - the producer was negligent
- They only need to prove that:
  - the product was defective
  - the defect in the product caused the injury.
- Directive applies to:
  - consumer products
  - products used at a place of work
  - all products are covered since 2000 including
    - primary agricultural products and games



Conditions: Product liability in the EU

1. A product must be involved
2. The product must be defective
3. The manufacturer must be identified
4. The damage suffered must be covered by product liability law
5. There must be a causal relationship between the damage and the product defect

Conditions relating to product liability

A manufacturer shall only be liable for a product where the following five conditions apply:

1. **A product must be involved;**
   It must first be established whether a product actually exists. According to the definition given under European and Swiss law, the product must be a movable.

   Movables also include
   - Movables which are incorporated into an immovable, such as a lift, even if the immovable it itself not covered by
     product liability;
   - Agricultural products such as cheese or wine;
   - Software, even if this is transferred via the Internet

   Movables do not include
   - Immaterial goods such as patents;
   - Services

- http://www.osec.ch/internet/osec/en/home/export/products/initial_information/een/prod_liability/product_liability_in_the_eu.html

Different Types of Software

Unusable Software

Software with hidden Bugs

Safety-Critical Software

Open Source Software

Known/Accepted Problem

- Zero-defect software does not exist (yet)
  - Implicit acceptance that software may fail or have bugs

Different Outcomes

- Software is not usable
  - Company users have problems
- Company customers have problems
- Company loses money
- Company customer lose money
- Software causes human injuries/fatalities

Safety-Critical Software

Examples

- Flight Control
- Nuclear power plant control
- Financial market
- London Ambulance Service
- Health related tools (radiation therapy)

UK

- No recovery if losses are purely economic

Open Source Software

Disclaimer of Warranty

"COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER. "

http://www.opensource.org/licenses/cddl1.php

Open Source Software

Limit of Liability

"UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTYS NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU."

http://www.opensource.org/licenses/cddl1.php

Good Practice When Developing Software

ACM Software Engineering Code of Ethics and Professional Practice

Software engineers shall commit themselves to

- making the analysis, specification, design, development, testing and maintenance of software a beneficial and respected profession.
- In accordance with their commitment to the health, safety and welfare of the public, software engineers shall adhere to the following Eight Principles:
  - 1. PUBLIC - Software engineers shall act consistently with the public interest.
  - 2. CLIENT AND EMPLOYER - Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.
  - 3. PRODUCT - Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.

www.acm.org/about/se-code

Good Practice When Developing Software

- 4. JUDGMENT - Software engineers shall maintain integrity and independence in their professional judgment.
- 5. MANAGEMENT - Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
- 6. PROFESSION - Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.
- 7. COLLEAGUES - Software engineers shall be fair to and supportive of their colleagues.
- 8. SELF - Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.

www.acm.org/about/se-code

IT Projects Failure

THE NATIONAL PROGRAMME FOR IT IN THE NHS (NPfIT)

NPfIT – Formally established in 2002

- Electronic care record of patients
- Links Hospitals – GPs
- Patient may have access to record on-line
- NPfIT is said to be the "world's biggest civil information technology programme"

Open Letter to the Health Select Committee

- Signed by 23 academics
- Express concern about risks of failure of project:
  - Price of IT project
  - Reliability of suppliers
  - Delays in delivery

Discussion with Director-General of NPfIT

On-going …

- London hospital experience delays after deployment of a patient record system
  - http://www.e-health-insider.com/News/4205/barts_still_struggling_with_cerner_crs
- Political condemnation
  - http://www.e-health-insider.com/News/4281/political_row_over_npfit:_london_on_hold

http://www.nhs-it.info/

NPfIT

The Open Letter to the Health Select Committee From NHS It Info

THE NATIONAL PROGRAMME FOR IT IN THE NHS (NPfIT)

The Select Committee may be aware of the concerns of health professionals, technologists and professional organisations about the £6bn NHS National Programme for Information Technology (NPfIT):

- The NHS Confederation has said "The IT changes being proposed are individually technically feasible but they have not been integrated, so as to provide comprehensive solutions, anywhere else in the world".
- Two of NPfIT’s largest suppliers have issued warnings about profits in relation to their work and a third has been fined for inadequate performance.
- The British Computer Society has expressed concern that NPfIT may show a shortfall of billions of pounds.
- Various independent surveys show that support from healthcare staff is not assured.
- There have been delays in the delivery of core software for NPfIT.

NPfIT

Concrete, objective information about NPfIT's progress is not available to external observers.

Reliable sources within NPfIT have raised concerns about the technology itself.

The National Audit Office report about NPfIT is delayed until this summer, at earliest; the report is not expected to address major technical issues.

As computer scientists, engineers and informaticians, we question the wisdom of continuing NPfIT without an independent assessment of its basic technical viability.

We suggest an assessment should ask challenging questions and issue concrete recommendations where appropriate, e.g.:

- Does NPfIT have a comprehensive, robust:
  - Technical architecture?
  - Project plan?
  - Detailed design?

NPfIT

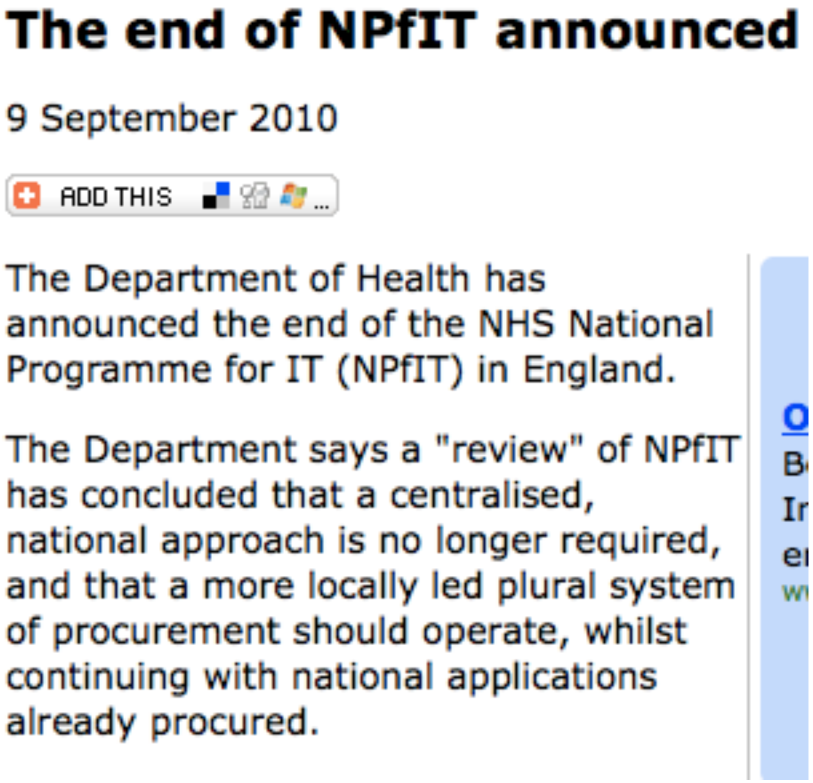
Have these documents been reviewed by experts of calibre appropriate to the scope of NPfIT?

- Are the architecture and components of NPfIT likely to:
- Meet the current and future needs of stakeholders?
- Support the need for continuous (i.e., 24/7) healthcare IT support and fully address patient safety and organisational continuity issues?
- Conform to guidance from the Information Commissioner in respect to patient confidentiality and the Data Protection Act?
- Have realistic assessments been carried out about the:
- Volumes of data and traffic that a fully functioning NPfIT will have to support across the 1000s of healthcare organisations in England?
- Need for responsiveness, reliability, resilience and recovery under routine and full system load?

NPfIT

We propose that the Health Select Committee help resolve uncertainty about NPfIT by asking the Government to commission an independent technical assessment with all possible speed. The assessment would cost a tiny proportion of the proposed minimum £6bn spend on NPfIT and could save many times its cost.

The End of NPfIT



The end of NPfIT announced

9 September 2010

The Department of Health has announced the end of the NHS National Programme for IT (NPfIT) in England.

The Department says a "review" of NPfIT has concluded that a centralised, national approach is no longer required, and that a more locally led plural system of procurement should operate, whilst continuing with national applications already procured.

http://www.bjhcim.co.uk/news/2010/n1009008.htm

What is a Contract?

An agreement between two or more parties

- legal or natural persons

Subject to:

- All parties intending to make a contract
- All parties being legally competent
- Must be a “consideration” – ie a “deal” for all parties

Software Contracts

- Bespoke systems
  - Fixed price
  - Time and materials
- Consultancy and Contract Hire
- Outsourcing and Offshoring
- Licence Agreements

Fixed Price Contracts for Bespoke Systems

Three parts to a typical contract

1. Short Agreement – specifies parties and states that anything written or said before is NOT part of the contract
2. Standard Terms and Conditions
3. Set of Schedules or Annexes specifying
   - What is to be delivered
   - When
   - What payments are to be made
   - Liability
   - …

What is to be Produced?

- Contract refers to the standard Terms
- Standard Terms refer to Requirements Specification
- Specification needs to set out detailed requirements of the client
- It should be complete, consistent and accurate
- However this is very difficult to achieve

Requirements Drift

- Requirements may evolve during project
- How can we incorporate changes?
- Contracts must provide for changing the specification including
  - Charging for making changes
  - Possibly amending performance specifications and related acceptance testing

What is Delivered?

Contracts are usually for an installed system and so can include:

- Source code
- Command files to build and install code
- Documentation of design and code
- Reference, training and operations manuals
- Software toolkit
- Training of systems staff, users etc
- Supply of test data and results

Ownership & Confidentiality

- Contracts should state legal rights of parties to finished work at the end of the project
- Usually the software house passes ownership of everything to the client

- Building IT systems for a client leads to both parties obtaining confidential information about each others businesses and business methods.
- Each party normally promises to respect the others confidential information

Payment Terms

Normally the standard terms & conditions will apply, typically:

- Payment within 30 days of invoice or if payment is delayed the contractor may terminate the contract or charge surcharge such as base lending rate + 2%

- Payment for a project will be staged – not least to pay the staff salaries - such as
  - Initial payment of 15%
  - Staged payments up to 65% of total projected cost
  - 25% on acceptance
  - 10% "retention" payable at end of warranty period

Delays and Changes - 1

- **Clients failing to meet obligations** on time lead to delays and additional costs for supplier
- Contract must provide for calculating cost of unrecoverable delays
- Typically Annex to contract will give day rates for staff and extra charges will be agreed at progress meetings
- Such payments are a frequent cause of legal disputes!

Delays and Changes - 2

- **Delays can be caused by Suppliers** and incur Penalty Charges
- Contracts will have delivery schedules
- Over-runs can lead to deductions from the fee of a cash sum per week
- Not often used because:
  - Suppliers not keen!
  - Suppliers will inflate the (fixed) bid price
  - Serious delays can lead to suppliers walking away since no further payments are due
- Over-runs reduce suppliers profit margins and this is usually sufficient incentive to deliver on time

Client’s Obligations

Contracts must specify what client must provide:

- Documentation of activities relevant to project
- Access to relevant staff
- Machine facilities, network links etc
- On site facilities and support

Project Management

Parties must agree on:

- Software supplier’s design and development methods
- Quality assurance methods to be used
- Progress meetings and reporting
- Project managers and client contacts

Acceptance and Warranty Period

- Acceptance procedures must be defined at the outset
- At start of Acceptance, client should provide a fixed set of acceptance tests and expected results
- Client cannot add extra tests later – thus delaying completion unreasonably - except by mutual agreement

- Warranty periods are normally 90 days – within which errors will be corrected free of charge
- Subsequent “maintenance” is usually on a time and materials basis

Other clauses

- Indemnity – each party may cause the other to infringe a third parties rights so both parties indemnify each other against any liability
- Termination – contracts may need to be terminated and conditions must be set in advance and cover finance, notice, ownership of the incomplete work
- Arbitration – commonly the Presidents of the BCS or IEE are named as arbitrators in contracts and both maintain lists of suitable experts
- Applicable Law – under which legal jurisdiction is the law written, eg English, Scottish, US states etc

Consultancy and Contract Hire

Contract Hire involves supplying staff to a client to work on a project managed by the client – "body shop". Similar to freelance or contract work.

Consultancy operates on a similar basis although they often specify a requirement. Issues to be covered in contracts are:

- Confidentiality
- Terms of reference
- Liability
- Ownership of final report

Time and Materials or Cost Plus

- Similar to fixed price bespoke contract but charged on hours worked

- Why would client accept such a contract?
  - Suppliers won’t tender for a fixed price
  - Work insufficiently specified
- Risk can be managed by having agreed milestones and termination clauses
- Risks are shared between client and supplier

Outsourcing and Offshoring

IT services have always been purchased from third parties, such as computer bureaux, software package suppliers, software houses (outsourcing)

In early 1990s some companies and government departments handed over the whole of their IT departments to third parties such as EDS and Accenture

Experience has been varied but many commercial companies discovered that

- Fast IT innovation gave them competitive advantage
- While outsourcing meant predictable running costs, development costs soared and
- Lead times for new developments were too long

Some organisations have brought their IT (all or some) back in house

Recently growing competition from Indian and other IT service suppliers moving complete IT functions out of the UK (offshoring)

Licence Agreements

When you buy software, you buy a copy of the software and the right to use it

Restrictions are placed on its use, for example:

- Single user licence, eg computer game
- Server licence for limited number of users
- Site licence to cover all users of system

Defective Software

Software almost invariably contains bugs

Suppliers try to limit their liability

Unfair Contract Terms Act 1977 prevents such a defence against death or personal injury

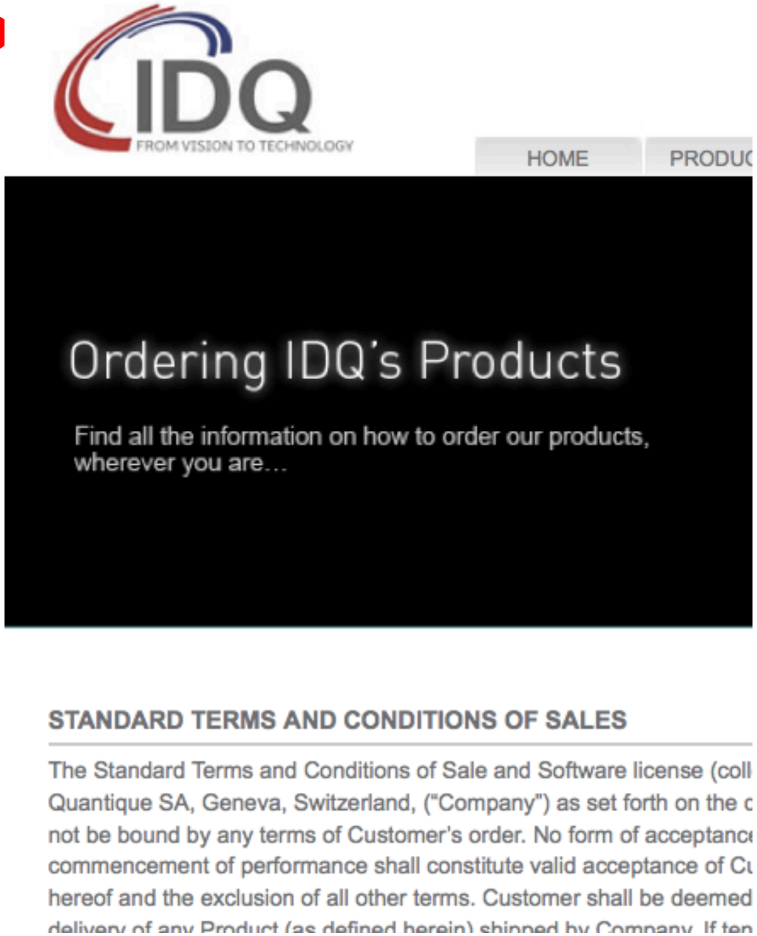
Defective Software

Sale of Goods Act 1979 and Supply of Goods and Services Act 1982 apply to software

Sale of Goods Act requires that a product is “fit for purpose” but applies to consumer sales only – not business sales

Supply of Goods and Services Act 1982 would apply to all sales but only requires “reasonable skill and care” which would be hard to disprove in court

Examples



Ordering IDQ's Products

Find all the information on how to order our products, wherever you are…

STANDARD TERMS AND CONDITIONS OF SALES

The Standard Terms and Conditions of Sale and Software license (coll
Quantique SA, Geneva, Switzerland, ("Company") as set forth on the c
not be bound by any terms of Customer's order. No form of acceptance
commencement of performance shall constitute valid acceptance of Cu
hereof and the exclusion of all other terms. Customer shall be deemed
delivery of any Product (as defined herein) shipped by Company. If ten

http://www.idquantique.com/ordering/terms-and-conditions.html

Summary

We have looked at:

- Contracts for the supply of software
- Considerations around the supply of software
- Legal use of software
- Liabilities and limit of liabilities

References

Franck Bott: Professional Issues in Information Technology, Ch.12, The British Computer Society, 2005.