Issues in Commercial Law of Interest to Software Engineering Educators

Conference on
Software Engineering Education & Training

February 2002, Covington, KY

Cem Kaner, J.D., Ph.D.

License and Disclaimer

- License: You may reproduce these slides, in whole or in part, for academic teaching and research use. You may not sell copies of them to anyone, other than students at or below your reproduction cost, without my written permission. Any slides that you reproduce must include my notice (Copyright (c) Cem Kaner, 2002) at the bottom of each slide, and this license notice and disclaimer must appear in your slide set. To contact me, send email to kaner@kaner.com.
- Disclaimer: These notes include some legal information, but you are not my legal client and I am not providing specific legal advice in the notes or in the course. Even if you ask me a question about a specific situation, you must understand that you cannot possibly give me enough information in a classroom setting for me to respond with a competent legal opinion. I may use your question as a teaching tool, and answer it in a way that I believe would “normally” be true but my answer could be completely inappropriate for your particular situation. I cannot accept any responsibility for any actions that you might take in response to my comments in this course. If you need legal advice, please consult your own attorney.

License Restriction Note

- On the disk, I have included copies of several documents, including statutes, court cases, and law review articles.
- The law review articles are copyrighted. I think that I am making “fair use” of them, using them here for teaching purposes. I cannot license you to use them in your courses, nor can I advise you whether a use in your course would be fair use or not. Make your own investigation and decision.
- The court cases are public domain but were downloaded from Lexis, which asserts a license right to prevent me from distributing them beyond my immediate classes. I cannot advise you as to whether it is appropriate to distribute public domain material to your class that was obtained via a commercial source with a restrictive license.
- You can solve the Lexis problem by downloading the cases yourself from another service provider. www.loislaw.com and www.versuslaw.com offer free trial periods and less restrictive licenses. If you sign up for them, you can also search for new cases (new cases are decided every day) on UCITA or computer law.
- It would be extremely unwise to copy materials from this disk onto a website open to the public. The fair use provisions of the copyright act will probably not protect you if you publish a document in a way that makes it available to everyone, not just to your students.

About Cem Kaner

- My career focus is the satisfaction (and safety) of software customers and workers.
- Prior to joining Florida Tech, I worked as a programmer, tester, writer, teacher, user interface designer, software salesperson, organization development consultant, as a manager of user documentation, software testing, and software development, and as an attorney focusing on the law of software quality. These roles provided many insights into relationships between computers, software, developers, and customers.
- Current employment
  - Professor of Computer Sciences, Florida Institute of Technology
  - Private practice in the Law Office of Cem Kaner
- Books
  - ***Lessons Learned in Software Testing: A Context-Driven Approach*** (with James Bach & Bret Pettichord, 2001).
  - ***Testing Computer Software*** *(*1988; 2nd edition with Hung Nguyen and Jack Falk,1993). This received the *Award of Excellence* in the Society for Technical Communication's *Northern California Technical Publications Competition* and has the lifetime best sales of any book in the field.
  - ***Bad Software: What To Do When Software Fails*** (with David Pels). Ralph Nader called this book "a how-to book for consumer protection in the Information Age."
- Education
  - J.D. (law degree, 1993). Elected to the American Law Institute, 1999.
  - Ph.D. (experimental psychology, 1984) (primary interests were in *measurement theory* and in *human factors*, the field concerned with making hardware and software easier and safer for humans to use).
  - B.A. (primarily mathematics and philosophy, 1974).
  - Certified in Quality Engineering (American Society for Quality, 1992-2001). Examiner (1994-95) California Quality Awards.
  - I also co-founded and/or co-host the Los Altos Workshops on Software Testing, the Software Test Managers' Roundtables, the Austin Workshops on Test Automation, the Workshop on Model-Based Testing, the Workshops on the Teaching of Software Testing, and the Workshops on Heuristic & Exploratory Techniques.
  - I'm a member of the American Bar Association, American Law Institute, American Psychological Association, American Society for Quality, Association for Computing Machinery, the Computer Law Association, Human Factors & Ergonomic Society, IEEE and IEEE Computer Society, Society for Technical Communications, Software Support Professionals Association, and the State Bar of California.

Context Setting

Why Teach Commercial Law to Students of the Computer Sciences?

Context: Different Areas of Law

- There are many different areas of law. For our purposes, I use a broad brush and paint them into 5 broad categories. This list is not complete (e.g. we ignore family law, etc.), but I find the categorization useful for thinking about CS Law / Ethics curriculum.
  - Public Safety
  - Property
  - Employment
  - Service-related duties
  - Commercial
- Orthogonal to these is the division of public (criminal) and private (a.k.a civil) law.
  - Example: Student shoots (wounds) a professor:
    - State sues student for attempted murder
    - Professor sues student for battery and its resulting harms (personal injury, pain, suffering, work time lost, etc.)

Context: Different Areas of Law

- **Public Safety**: *Governs risks of personal injury and physical damage to tangible property.*
  - Private law includes: intentional torts (like battery), negligence resulting in injury or property damage, and products liability (injury or property damage caused by a defective product).
  - Public law includes crimes of violence and other wrongful acts (or wrongful failures) to act that result (or reasonably could be expected to result) in personal injury or destruction of physical property.
  - NOTE: Tort law also covers some non-violent harms to a person, such as slander, libel, and various intrusions on a person's privacy.
- ***Public safety issues stimulate good discussions in class. They've also been described as prime concerns underlying the need for professionalizing software engineering.***

Context: Different Areas of Law

- **Property**: *Governs rights of the owner of property, and of users, buyers, lenders, lessors, licensors. The focus of this area is on the rights, responsibilities, and opportunities (e.g. ways to contract) that arise out of the specific type of property.*
  - Private law includes: real property (real estate) and intellectual property (copyright, trademark, patent, trade secret)
  - Public law includes zoning, the rules that establish intellectual property (these forms of property are relatively new and are created by statutes), and crimes involving misappropriation or criminalized infringement of property rights.
- ***We are living in an economic revolution. Intellectual property provides an increasing proportion of our national wealth. There are strong (conflicting) interests in tailoring the law's allocation of rights and wealth and so this law is undergoing rapid change and intense public discussion.***

Context: Different Areas of Law

- **Employment**: *Governs duties of employees and employers to each other.*
  - Private law includes: employment contracts (including non-competition clauses, confidentiality clauses, etc.)
  - Public law includes wage-and-hour laws and many other regulations.
- ***Most high-tech workers are required to sign employment contracts or consulting contracts. These are often broad and over-reaching. Many students appreciate the opportunity to review sample contracts and to discuss how to identify and negotiate the most controversial clauses.***

Context: Different Areas of Law

- **Service-related duties**: *A service contract is primarily governed by commercial law. But some service providers are required to perform their contract at a higher level of skill, diligence or integrity than an average person, because the provider is treated specially in law, e.g. as a licensed professional or a practitioner of some skilled trades.*
  - Private law includes: malpractice
  - Public law includes statutes regulating the professions, etc.
  - Quasi-public law includes professional societies' disciplinary boards.
- ***Figuring out what would and what would not constitute software engineering malpractice is a daunting task. In think students who are subject to malpractice risk would be disserved by a software engineering curriculum that did not focus at least a semester on "professional responsibility" (aka malpractice avoidance).***
- Articles included:
  - Kaner (1996) Computer Malpractice.
  - Kaner (2001) Software Engineering as a Profession.

Context: Different Areas of Law

- ***Commercial***: *Governs promises made in the course of doing business.*
  - Private law includes: contracts governing any type of commercial transaction (e.g. sales, leases, licenses), potentially subject to limits or revisions imposed by the State's consumer protection rules.
  - I include the law of (commercial) fraud within this broad area, even though fraud is handled by the courts as a tort.
  - Public law includes fraud (criminal prosecution), consumer protection regulations that might be enforced by the government, antitrust law, and other laws that limit any party's commercial freedom

So, Why Teach Commercial Law?

- Commercial law provides the legal context for most of our discussion of what is acceptable or unacceptable in the marketplace.
  - *COTS*: think of defects, warranties, advertising. *Custom software*: think of overruns, scope changes, disputes over defects and design disagreements.
- Commercial law will be the primary body of law governing many of the work products made by our students.
- Commercial law poses dangerous traps for the unwary manufacturer, vendor, and service provider (roles that our students will often play).

Strategy

A Structure that I'm Trying Out for Teaching this Material

Challenges of Teaching Commercial Law

- Knowledge and culture of the students
  - American students typically have incorrect impressions of their rights to good products and services and of their power when they are in a dispute with a seller. They live with a surprising collection of myths, wishful thoughts, overgeneralized conclusions drawn from political/economic propaganda, and not-yet-matured political viewpoints.
  - Many non-American students don't understand our culture our basic legal rules.
- We have entered a period of enormous controversy (often quite bitter) about the foundations of current commercial law

Teaching in the Face of Policy Conflicts

- Start with a question
  - For example, *What standard of performance (e.g. defect level) should we require of COTS manufacturers?*
- Consider focusing it with a fact pattern
- Dig up background information
  - What statutes currently apply to this question?
  - What case law currently applies to this question?
  - What solutions are in play in other industries?
  - What is the alternative proposal?
  - What is the history of this controversy?
- Facilitate the discussion
  - OK, so *what standard **should** we apply?*

Background for the Discussion

Background Notes on U.S. Commercial Law

Background Notes on U.S. Commercial Law

- U.S. law is multi-tiered:
  - Federal
  - State
  - County
  - City
- The federal government is granted under the constitution a limited set of powers. Anything not granted to the feds belongs to the states.
- Within its scope, federal law overrides (preempts) inconsistent state law.
- In the event of dispute, the Supreme Court decides what powers are federal.

Background Notes on U.S. Commercial Law

- Commercial law generally belongs to the states.
- Interstate commerce belongs to the feds. When inconsistencies among state laws interfere with interstate commerce, the federal government can create new laws as needed to relieve commerce (even if this means that the feds now take over part of an area of law that belonged to the states).
- In the 1860's, the United States government waged war against breakaway states. Slavery was one of the key disputes of the war. Note that this is a dispute over the scope of a state's authority to define its own commercial laws (in this case, to make and keep legal the sale and ownership of a human).
- Shortly after the war ended, the states created the National Conference of Commissioners on Uniform State Laws (NCCUSL). NCCUSL first met in Montgomery, Alabama but now meets all over the country.

Background Notes on U.S. Commercial Law (NCCUSL)

- Mission of NCCUSL is the creation of “Uniform State Laws” – laws that are well researched, carefully crafted, and acceptable to such a broad portion of society that they are likely to be adopted in every state.
- NCCUSL drafts laws in open meetings. In practice, these are poorly publicized and are attended primarily by representatives of the industries with the biggest stake in the result. Depending on the skill of the committee chair and the Reporter (senior author of the bill), the NCCUSL process shines or fails when there are serious disagreements.
- Once a committee approves a draft bill, it goes to NCCUSL as a whole for approval. If NCCUSL approves it, the bill goes to each state legislature, and each decides whether or not to adopt the bill. The bill also goes to the American Bar Association for review. The review is influential with the state legislatures.
- NCCUSL is made up of about 300 voting members, most of whom are appointed by the governor of their state.

Background Notes on U.S. Commercial Law (NCCUSL)

- Mission of NCCUSL is the creation of “Uniform State Laws” – laws that are well researched, carefully crafted, and acceptable to such a broad portion of society that they are likely to be adopted in every state.
- NCCUSL is made up of about 300 voting members, most of whom are appointed by the governor of their state.

Background Notes on U.S. Commercial Law (Common Law)

- America follows the “common law” model. Statutes go so far in specifying the law, but the courts have to apply the law to the facts, often crafting new law as they interpret what the legislature intended, or (if an issue has never been tried before in the state, and is not the subject of directly relevant statutes), what this legislature would intend if it considered the situation at hand.
- “Common law” is judge-made law.
- A judge sets a *precedent* if her decision will be followed by other judges in the future. Precedents are coercive. Subsequent judges “have to” follow them. Typically a precedent is created by an appellate level court, and it is coercive to all judges within the scope of that appellate court. A judge *can* make an inconsistent ruling, but if he does, the appellate court will probably reverse it, reinstating their old rule.

Background Notes on U.S. Commercial Law (Common Law)

- Hierarchical relationships are fairly clear, but what about horizontal ones? Imagine if you were a judge, trying a case involving a specific paragraph of the Uniform Commercial Code, and discovering that no judge in your state has ever issued a ruling that interpreted the meaning of that paragraph. By definition, you have no precedents to rely on.
- To decide what to do, you will probably read Opinions (judges document their decisions in Opinions) from other states or even other countries whose laws aren't too different from yours. Digging up and analyzing those opinions is very complicated and time-consuming. Judges differ in their time and ability to devote to this type of research.
- The *American Law Institute* was founded to research the common law and provide authoritative summaries of it.
- The American Law Institute is another influential, non-partisan legislative drafting body, more focused on common law than statute.

Background Notes on U.S. Commercial Law (ALI)

- The *American Law Institute* is another influential, non-partisan legislative drafting body, more focused on common law than statute. It was founded to research the common law and provide authoritative summaries of it.
- The main work products of the ALI are the *Restatements*, such as the *Restatement of Contracts*. A Restatement walks through a body of law, issue by issue. For each issue, the Restatement identifies and discusses the opinions that have addressed the issue. The Restatement explains the history of a given rule, points to scholarly research and cases, and counsels the reader as to what stand is the better stand or the dominant one.
- The ALI also co-drafts some legislation with NCCUSL, including the Uniform Commercial Code, which was probably the most successful legislation in U.S. history.
- The ALI has about 3000 members, primarily judges and tenured law professors, plus senior partners from big firms, and a few independents. I was elected to ALI in 1999 in recognition of my work on computer law.

Background Notes on U.S. Commercial Law (ABA)

- The *American Bar Association* is a large professional society, like ACM or IEEE.
- The ABA divides into Sections (like SIGs) which in turn appoint task forces (committees, working groups). These groups might study an area of law and produce a summary of the current state of affairs or a set of recommendations for legal practice (memos to the lawyers rather than to the judges or legislators). Or they might draft "model" legislation, which is submitted to the state or federal legislature. Or issue a report like a restatement, as an advisor for judges.
- The ABA also reviews any proposed uniform laws from NCCUSL. ABA is very influential. A decision to not endorse by ABA will harm the proposed law's credibility.
- ABA working groups are open to ABA members who belong to the Section that sponsors the working group. Any ABA member can join (almost) any Section.
- As a result, it is possible to "stack" a working group with representatives of one industry's interests dominating the meeting rooms.

Background Notes on U.S. Commercial Law

- NCCUSL, ABA, and ALI were significantly involved in the development of the statute called the Uniform Computer Information Transactions Act (UCITA).
- UCITA attempts to govern almost all transactions in computer-related “information” (“transactions” include all types of real or implied contracts)

  Each is playing a significant role in the development of new commercial laws and the interpretation of current laws.

Contract Fundamentals

- A contract is an *agreement* between two or more people (or companies) that creates obligations to do or to provide particular things.
  - In many cases, there is no agreement-creation process. Instead, we talk of the *voluntary assumption of an obligation* as the basis of the contract.
- A software contract can involve
  - *goods* (such as a program bought at a store)
  - *services* (such as custom programming)
  - some mix of the two (such as a program that comes with a maintenance contract).

Contracts: Uniform Commercial Code

- Article 2 of the Uniform Commercial Code is called the Law of Sales.
- It is the primary law governing contracts for sale of goods in USA.
- Sale of packaged software has been consistently treated by the courts as a sale of goods under Article 2.
- Sale of custom software is a sale of services, not covered by the UCC, though several courts have applied the UCC anyway.
- The UCC is maintained and updated by NCCUSL and ALI
- Some legislation and process notes:
  - The full text of the UCC is on-disk as **t42a-p1.htm** (this is Connecticut's implementation, but I believe it is fairly standard.) Sorry for the file name, but that's Connecticut's file name and there are many, many internal hyperlinks to a file of that name.
  - Alces, Peter & David Frisch (1998), Commercial Codification as Negotiation. *See alces.rtf*.
  - Hillman, Robert. (1999) The "New Conservatism" in Contract Law and the Process of Legal Change. *See hillman.rtf*.
  - Towle, Holly K. (1999) The Politics of Licensing Law. *See towle.rtf*

Traditional Terminology of Contract Law

- Offer
- Counter-offer
- Acceptance
- Consideration
- Battle of the forms
- Warranty
- Modification
  - The pre-existing duty rule
  - Material vs. non-material modifications
- Oral contracts are valid (except under statute of frauds)
- Contracts by conduct are valid
- Remedies

Uniform Commercial Code

- Interesting features of Article 2
  - Gap fillers, and implied terms
  - Perfect tender rule
  - Implied warranty of merchantability
  - Implied warranty of fitness for a particular purpose
  - Battle of the forms rules:
    - contract by conduct
    - forms as proposals for modification
  - Modification rules: material vs. non-material modifications
  - Some rules apply only to merchants

Uniform Commercial Code

- *Implied warranty of merchantability:* The product will be reasonably fit for ordinary use and will conform to claims on the packaging. This is a fundamental assumption of most deals, and if it is disclaimed, the disclaimer must be conspicuous.
- *Express warranty:* The product will conform to any statement of fact from the seller to the buyer about the product that becomes part of the basis of the bargain. It will also conform to characteristics shown in a product demonstration.
- *Perfect Tender Rule:* Customer can reject the product for defects that show up during an initial inspection of the product.
- *Battle of the Forms—Whose Terms Govern?* A sale often involves form contracts that neither side reads. The buyer sends a standard-form purchase order and the seller sends a standard-form acknowledgement, with conflicting terms. Article 2 provides a difficult, fact-specific set of rules. Often, a final set of terms from one side is really (or will be treated as) a proposal for a modification to the contract. The Article 2 rules were painstakingly tailored to be neutral (buyer vs. seller).

Uniform Commercial Code

- *Modifications:* Post-sale modifications are OK, but other party need not agree to them. Material modifications don't become part of the contract unless the other party does agree. ("Material" = "Important.")
- *Remedies:* Incidental and consequential damages.
  - Incidentals—costs associated with reporting the defect, returning the product, calling for support, etc.
  - Consequentials—all other costs/losses caused by the defect (or breach of contract).
  - The contract can limit remedies, but must provide a "minimum adequate remedy."

Magnuson-Moss Act and Consumer Statutes

- Magnuson-Moss Warranty Improvement Act (Mag-Moss)
  - Passed in response to widespread abuse (deceptive practices and confusing practices) by sellers.
  - Requires vendors to make their warranties available pre-sale.
  - Defines some common terms used in warranties.
  - Forbids disclaimer of implied warranty if any written warranty is offered.
- Song-Beverly Act (CA) and other State Laws
  - Extend Mag-Moss, for example by saying that disclaimer must actually be provided to consumer pre-sale. Some of these statutes ban disclaimers of implied warranties in the sale of new merchandise.

Contracts: UCITA

- Intended to update the UCC, to handle software more sensibly:
  - Governs all contracts involving software and digitally stored information.
  - Unified framework for software and digital content.
  - Unified treatment of products and services
  - Clarify when / where a contract is formed, how formed, and how modified.
  - Clarifies the rules governing shrinkwrap and clickwrap licenses
  - Clarify the rules for informing customers of contract terms, including (especially) warranty terms, remedies, use restrictions, and transfer restrictions.
  - Enables use restrictions, a concept not available in contracts for sales of goods. More generally, allows more flexible licensing terms, which may affect software pricing
  - Strengthens protections against piracy and use outside of the reasonable expectations of the vendor
  - Opt-in clauses can bring in goods sold with software.
  - Minimizes risk of frivolous lawsuits against vendors
- Current draft of UCITA: www.law.upenn.edu/bll/ulc/ulc.htm. Current draft and amendments on-disk as ucita2001.rtf and ucitaamend.rtf
- In our terminology, UCITA adopts a Commercial Allocation of Risk model.

Contracts: UCITA

- Passed as law in Virginia and Maryland.
- Three states (Iowa, two others) have passed a “bomb shelter” law that invalidates contract clauses that say that UCITA governs contracts.
- This is highly controversial but extremely influential law.
- UCITA was first developed in committees of the American Bar Association, starting in the late 1980’s.
- NCCUSL and ALI picked up the project in 1991, initially folding it into a maintenance project for Article 2 of the UCC (the Article 2 revision project).
- The Article 2 revision committee couldn’t reach agreement on the policies being proposed for software transactions. Ultimately, the committee split into an Article 2 (law of sales) committee and an Article 2B (law of software licenses) committee.
- The Article 2B project ended when the American Law Institute withdrew, citing concerns that Article 2B was trying to write around federal Copyright law and that it was making fundamental changes in contract law.
- NCCUSL renamed the bill’s name to UCITA and went forward on its own. This is the first time that ALI and NCCUSL have split.

Politics of UCITA

- Typical proponents:
  - Software publishers (e.g. Microsoft, Symantec, Adobe, Oracle)
  - Online service providers (e.g. AOL)
  - Computer manufacturers (e.g. Dell, and I believe, Gateway)
  - Database publishers (West / Lexis / NASDAC)
  - CitiBank, Daimler Chrysler
- Typical opponents
  - Consumers
  - Librarians, Infoworld
  - Insurance companies, Caterpillar, Boeing, Sun Micro, SIM, CCIA
  - Staff members of the Federal Trade Commission, 33 Attorneys General
  - American Intellectual Property Law Assoc and IP section of the NY City Bar Association
  - ACM-USA, IEEE-USA, ICCA, CPSR, ASQ, SEI, professional societies and trade groups representing software engineers.

ALI's Withdrawal from UCC 2B

- ALI passed resolutions in May 1997 and 1998 calling for fundamental revision of Article 2B. It withdrew from Article 2B in 1999.
- The 1997 resolution attacked the Article 2B approach to intellectual property, and urged the drafters to conform Article 2B to federal intellectual property policies, as expressed in the federal laws governing copyrights and patents.
- The 1998 resolution focused on customers and competition. This is from the supporting memo to the May 1998 ALI resolution (Braucher and Linzer):

  > "The Draft reflects a persistent bias in favor of those who draft standard forms, most commonly licensors. It would validate practices that involve post-purchase presentation of terms in both business and consumer transactions (using "shrink-wrap" and "clickwrap"), undermining the development of competition in contingent terms, such as warranties and remedies. It would also allow imposition of terms outside the range of reasonable expectations and permit routine contractual restrictions on uses of information traditionally protected by federal intellectual property law. A fundamental change of approach is needed."

So, We Have a Policy Debate

- The policies (risk allocation, notice of terms, etc.) underlying UCITA and UCC Article 2 are quite different.
- Since 1995, courts have begun applying UCITA thinking to software transactions, even though they give the appearance of deciding the case under Article 2.
- The Article 2 revision committee has not adopted the UCITA rules, which is creating enormous tension (reaffirmation of the old Article 2 by a legislature will cast some newer Article-2-ish software decisions into doubt).

So, which of these approaches provides the best answers for software, or is there some better third way?

Background for the Discussion

Design Issues in the Development and Evaluation of Commercial Law

Commercial Law

- A body of laws forms a system. An individual law is like a feature that interacts (often, with excessive coupling) with other features.
- It is interesting (for me, very useful) to evaluate proposed new laws in terms of the desirable design characteristics of the body of law they are proposed to join.
- Commercial law provides a foundation for commerce.
  - The essence of commercial law is that it exists to facilitate commerce.
- Five key factors that facilitate commerce are:
  - Clarity
  - Uniformity
  - Stability
  - Freedom of contract, and
  - Appropriate allocation of accountability

Commercial Law

- Clarity

  Makes it easier for parties to understand the rules themselves, or with relatively simple instruction. They spend their time and money on commerce, rather than on lawyers.

- Uniformity

  Buyers and sellers avoid the misunderstandings that are caused by conflicting rules.

Commercial Law

- Stability

  allows parties time to develop a thorough understanding of the rules and to rely on what they already know. Buyers and sellers do not have to hire lawyers to reanalyze the same old issues. They know what risks they're taking and who has to buy the insurance. Laws that tilt too much toward one side of a transaction do not promote stability. If one side is shabbily enough treated, it will bring political pressure to bear to get a fairer shake. Maneuvering for significant short-term advantages might be successful over the short term, but the backlash is inevitable.

Commercial Law

- Freedom of Contract

  Allows the parties to make the agreements that they choose to make. Of course, there are assumptions underlying this freedom.

  - Access to information
  - Ability to bargain
  - Ability to understand the bargain
  - Public policy limitations

An essential aspect of economic freedom is the freedom to make a bad deal. If the government protects you from all your (potential) mistakes, it eliminates your power to take risks.

Appropriate Allocation of Accountability

- If a product is defective, who should pay how much? There are several approaches to risk/accountability allocation, depending on which policies are most important to serve.
- Over time, social policy choices change. That is, societies gradually rotate the policies over time.

Accountability: Laws are Subject to Dynamic Balances

Erroneous Classifications Will Happen.

- A decision rule that is less complex than the situation being classified will make mistakes.
  - Sometimes buyers will lose when they should win.
  - Sometimes sellers will lose when they should win.
- Both sides will have great stories of unfairness to print in the newspapers.

Accountability: Laws are Subject to Dynamic Balances

How Should We Bias the Law?

- Given that some level of error is inevitable, our policy choices will strongly influence *which errors are more likely to be made.*
  - Should sellers have to accept returns of good products?
  - Should buyers have to tolerate defects?
  - Who should pay for the consequences and costs of "user errors"? Why?

Accountability

FAULT-BASED LIABILITY

- **Theme**: It's your defect. You should pay for it.
- **Challenge:** how do we define "defect"?
  - Failure to meet specification?
  - Failure to satisfy reasonable customer expectations?
  - Failure to protect the user from a reasonably foreseeable risk of injury?
- **Problem**: At what point do we create a lottery that randomly takes companies out of business?
  - Can you make zero defect software? No?
  - In a purely fault-based system, can you stay in business?

Accountability

TECHNOLOGICAL RISK MANAGEMENT

- **Theme**: You should have done it better.
- The seller/developer is liable when external failure cost < total cost of defect to society
  - Encourages improvements
  - The improvements can drive up cost to the customer.
- **Problem:** in the extreme, this can become a huge tax to protect fools from their own recklessness.

  - See the discussion of quality-related costs for definition of external failure cost.

Accountability

COMMERCIAL MANAGEMENT

- **Theme**: Freedom of contract. Allocate risk by contract.
  - Makes perfect sense when there is genuine power to negotiate.
  - What about non-negotiable situations?
  - The system is generally indifferent to quality. It provides no incentives or disincentives for quality—the contracting parties work that out.
- **Problem**: When there is significant bargaining disparity, the more powerful party can (probably will) structure the contract to protect itself when quality is bad. As a result,
  - The overly powerful vendor has little incentive to improve products. Customers just have to put up with the problems.
  - The overly powerful buyer is likely to drive several small-to-medium vendors out of business by "forcing" them to guarantee performance beyond the reasonable quality expectations for the product.

    - (In law, the "parties" to a contract are the people or organizations who agreed to the contract or who are otherwise bound by it.)

Background for the Discussion

Customers' Costs

and

Quality-Related Litigation

Bad Software Data

- Up to this point, students find the material pretty theoretical. Early in the class, people are willing to be patient with me, but if I don't get a rise out of them soon, they are lost.
- For many students, the notion that customers should be able to sue for bad software is surprising (even offensive). They raise several objections, such as:
  - Software problems aren't all that serious. They inconvenience people but if users weren't stupid, there wouldn't be many problems.
  - The contract says you can't sue the company. If you weren't willing to accept that, you shouldn't have signed the contract (or clicked OK).
  - In my country, you could never sue people for something like this. You Americans are very strange.
  - People in the United States sue everyone all the time. It is ridiculous. It is destroying our industries. We need to protect companies from irresponsible customers.

Bad Software Data

- The next slides provide some concrete data on commercial dissatisfaction / costs.
- This doesn't address all of the student objections. We'll talk about some of the other issues later. But it gives a grounding and wakes up some students.
- The data quoted are up to date circa 1997-1998. The cost of collecting these numbers was enormous and self-funded, and I have only updated a few of them.

Basic Quality Engineering: Quality-Related Costs

- The Cost of Quality is the total amount the company spends to achieve and cope with the quality of its product.
- This includes the company's investments in improving quality, and its expenses arising from inadequate quality.
- The primary goal of quality engineering is often described as minimization of quality-related costs.

Basic Quality Engineering: Quality-Related Costs

- Prevention

  Cost of preventing software errors, documentation errors, and any other sources of customer dissatisfaction

- Appraisal

  Costs of looking for defects (all types of inspection and testing).

- Internal Failure

  Costs of coping with errors discovered during development.

- External Failure

  Costs of coping with errors discovered, typically by your customers, after the product is released.

- Total Cost of Quality = Prevention + Appraisal + Internal Failure + External Failure costs.

Categorizing Quality-Related Costs

Prevention	***Appraisal***
• Staff training • Requirements analysis • Early prototyping • Fault-tolerant design • Defensive programming • Usability analysis • Clear specification • Accurate internal documentation • Pre-purchase evaluation of the reliability of development tools	• Design review • Code inspection • Glass box testing • Black box testing • Training testers • Beta testing • Test automation • Usability testing • Pre-release out-of-box testing by customer service staff
Internal Failure	***External Failure***
• Bug fixes • Regression testing • Wasted in-house user time • Wasted tester time • Wasted writer time • Wasted marketer time • Wasted advertisements • Direct cost of late shipment • Opportunity cost of late shipment	• Technical support calls • Answer books (for Support) • Investigating complaints • Refunds and recalls • Interim bug fix releases • Shipping updated product • Supporting multiple versions in the field • PR to soften bad reviews • Lost sales • Lost customer goodwill • Reseller discounts to keep them selling the product • Warranty, liability costs

Counting the Costs – Summary and Risks

- Software developers and publishers make cost/benefit tradeoffs in determining how they design reliability into their software and how carefully they test and fix it.
- Those tradeoffs tend to focus on the vendor's costs and not on the customer's costs.
- It is too easy to focus on easy-to-measure failure costs, such as Technical Support costs.
- To deal with these costs, many companies sell support time and turn their bugs into a profit center. In this and other ways, the numbers that were easiest to count (tech support or warranty cost) can be driven down without improving the product or the customer's experience.
- Lost sales from repeat-potential customers probably cost much more than tech support costs but we often ignore them because:
  - They are hard to measure.
  - Customers sink a heavy investment in some products and face huge transition costs if they choose to switch to a replacement.
  - The competition in many software markets is quit thin.
- Changes in the legal structure that reduce vendor liability for defects will make it cheaper for the vendor to provide lower quality products or services.

Do You Remember the Pinto? (And the Mustang?)

- These were the estimates at Ford:
  - External Failure Costs = $49.5 million
    - 180 burn deaths $200,000 each
    - 180 serious burn injuries $67,000 each
    - 2100 burned vehicles $700 each
  - Total Costs to Repair = $137 million
    - $11 per vehicle
- External failure costs are cheaper than repair, therefore ship it. Right?
- This is classical quality/cost analysis, so what's the problem? (Ford had to pay millions in punitive damages. But why? If this is normal quality-cost analysis, why should society punish Ford?)

- These cost estimates were published in Keeton, W. P., Owen, D.G., Montgomery, J. E., & Green, M.D. (1989, 2nd Ed.) *Products Liability and Safety, Cases and Materials,* Foundation Press, page 841 and Posner, R.A. (1982) *Tort Law: Cases and Economic Analysis*, Little Brown & Co., page 225.

The Blind Spot of Quality / Cost Analysis

- Quality/Cost analysis teaches the company to focus on its own costs, to minimize the sum of quality-related costs that are paid by the company.
- ***This analysis ignores externalized failure costs -- the costs absorbed by the customer.***
  - (See my paper, Quality / Cost Analysis: Benefits & Risks.)
- ***What about*** the quality-related costs that are paid by the customer?

- ***Question for discussion: Can any cost-control scheme be ethical if it limits our thinking about risks to our customers? Why or why not?***

Customers Have Failure Costs Too

Seller: external costs	***Customer: failure costs***
These are the types of costs absorbed by the seller that releases a defective product.	***These are the types of costs absorbed by the customer who buys a defective product.***
• Technical support calls • Preparing answer books • Investigating complaints • Refunds and recalls • Interim bug fix releases • Shipping updated product • Supporting multiple versions in the field • PR to soften harsh reviews • Lost sales • Lost customer goodwill • Reseller discounts to keep them selling the product • Warranty, liability costs • Gov't investigations	• Wasted time • Lost data • Lost business • Embarrassment • Frustrated employees quit • Demos or presentations to potential customers fail because of the software • Failure during tasks that can only be done once • Cost of replacing product • Reconfiguring the system • Cost of recovery software • Cost of tech support • Injury / death

Irrational Myths About Customer Dissatisfaction

- Myth: publishers are helpless to prevent defects.
  - *In mass-market products, most calls involve known defects.*
- Myth: most calls for help reflect customer ignorance or fault.
  - *Kaner / Pels data (desktop publishing application): 50% of calls could have been prevented with cheap fixes.*
- Myth: no one uses documentation.
  - *Dataquest -- 85% of people in trouble solve their own problem*
  - *Kaner's data (financial application) -- 88% of callers said they checked the docs first and could identify the weakness in the doc that led them to give up and call for help.*
- Myth: investments in support aren't good business.
  - *SoftLetter on MS's $500 million investment in support:* ***"Despite lots of wishful thinking to the contrary, spending money to upgrade a company's service reputation remains a lousy investment."***

Another Slide on the Soft-Letter article

- Customer dissatisfaction with quality significantly reduces a company's sales, but several (in my experience, most) companies ignore the dissatisfaction-associated revenue risks because they don't know how to estimate their magnitude.
- The degree to which people underestimate long-term effects is illustrated by the following example.
  - In the early 1990's, Microsoft spent $500,000,000 bringing its customer support from blecch to world class. But (1996) customer perceptions still rank MS near average as a support provider. Therefore, there might not be an obvious immediate payoff in sales volume. Result--a leading and influential newsletter on software publishing operations concluded,

    "Despite lots of wishful thinking to the contrary, spending money to upgrade a company's service reputation remains a lousy investment."

  Managing and consulting in Sili Valley during this period, my sense was that MS Office took over its market partially because competitors committed customer satisfaction suicide.

Bad Software: Defects

- Complete testing is impossible. No one ships bug-free software. (And if they did, they wouldn't know it.)
- In software, we routinely ship products with many known defects. Publishers use cost/benefit analyses to determine which bugs to fix. The most responsible publishers pick the bugs they ship more carefully and deal with complaining customers more responsibly.
- Software test groups pride themselves on the low rate of "surprises" in the field. (A surprise is a bug that the customer finds that had not already been found in testing.) For example, Roger Sherman, the former Director of Software Testing at Microsoft reported that over a 2-year period, typical MS products yielded only 2 surprises.
- Immense pressure to ship products quickly: The 4th competitor to market probably gets less than a 3% share.
- High cost of entry for new publishers' products. Over-investment in reliability or too-high a risk of liability will kill startups.
- Watts Humphrey and colleagues report products with nearly zero coding errors discovered in the field. The state of the art is advancing. (But the company has to be willing to spend the money to apply the advanced methods.)

Recipe for Legal Trouble

1. Mis-set customer expectations
2. Add defects
3. When customers call for support:
    (a) waste the customer's time
    (b) lie
    (c) blame it on the customer
    (d) don't solve the problem
    (e) be rude to the customer

Likely results?

    (a) Loss of customer loyalty
    (b) Lawsuits

Bad Software: Mis-set Customer Expectations

- The Canadian government (Industry Canada, Competition Bureau) recently completed a study of the claims made on the packaging of consumer software. Here is the bottom line:
  - "Over 2000 claims were evaluated during this survey. Overall, some 163 or 8.1% of all claims evaluated were potentially false or misleading. While this may appear to be a small percentage based on the number of claims evaluated, these incorrect claims represent 65% of all the software titles tested."
- For detail, http://strategis.ic.gc.ca/FBP and search for "software".
- 50% of software publishers don't give their manuals to the test group for testing. (Savings: about 15 minutes labor per page.) (Risk: In most states, the manual creates express warranties. The product must conform to all of the manual's "statements of fact". The manual provides evidence of breach of contract.) (Research from Customer Care Institute.)

Bad Software: Mis-set Expectations

- Albert Stark described problems that software support staff encounter when they try to buy and install problem management systems. Support staff provide an interesting example, because they're usually pretty talented at making things work.
- Stark points out that:
  - "The system will not do everything promised."
  - "System functionality is typically overstated."
  - "You'll need to purchase additional modules to get the functionality you need."
  - "Features you need are scheduled for a future release."
  - "The out-of-box reality is less than expected."
  - "You'll need to purchase additional hardware."
  - "The software will be more complex than it appeared during the sales cycle."
  - "System customization will not go smoothly" even though "Vendors can make customization look easy."
- In a parallel session at the same conference, the speaker asked publishers' technical support staff how many of them would trade in their problem management system if they could. Over half the attendees raised their hands.

Bad Software: Pressure on Support

- Skyrocketing support costs: staff ratios rose from 1:12 to 1:7 over 1990-1996. In 1996, 200 million calls to tech support.
- Software companies spend about $3 per minute providing support for PC-based products, and $5 per minute (or more) for UNIX and mainframe products. This averages $25-40 per call. With outsourcing (and different cost estimation structure), costs are often estimated at $8 per call now.
- Some companies have pushed many complainers to the internet. Net support is not free, but is less expensive for moderately high volumes of calls or for highly repetitive calls. In these companies, handling the issues raised in live calls cost as much as $150 to $400 per incident (averages reported at a 1999 Support Services Conference). Increasingly complex hardware/software configurations drive up support calls and costs. Complaints involving software / hardware from more than one vendor take 3 to 18 times as long to resolve.
- For more data, (from Prognostics, SSPA, Softbank, etc.), see Bad Software (my book) or http://www.badsoftware.com/stats.htm. (Additional recent data in *Software Engineering & UCITA*, which comes with these notes.)

Bad Software: Genuinely Bad Support

- Companies routinely deny their defects during calls for support (even known defects):
  - we've never heard of that
  - it must be your video card
  - it must be you
  - it's a feature
- Doing research for *Bad Software*, David Pels and I dug up bug reports on BugNet and then called publishers to complain about those bugs. We *always* got denials. After getting past those, we *still* often heard these other excuses.

Bad Software: Bad Support

- Software companies have started charging for support. $3 per minute or $35-95 per call (or incident) are common. Some companies charge even in the event of known bugs. Quality / cost pressures from support cost go away.
- Computer-related complaints made Better Business Bureau's top 10 for 1995, even higher than used car dealers. We did worse in 1996. (The BBB's data for 1997 merged computing with consumer electronics, making comparisons with the 1995 and 1996 data difficult. The combined totals yield higher ranks (more complaints), of course.)
- Customer satisfaction with software technical support has declined for ten straight years. In 1997, Prognostics Corp. claimed the trend has leveled off. Softbank still cites a decline.
- Cross-industry study of call hold times: Complaining software customers left on hold for longer than any other industry studied, even longer than airlines and government offices.

Bad Software: Bad Support

- Back to those 200 million calls for support.
  - Software customers spent over 3 billion minutes on hold. These are longer complaint hold times than other industries.
  - This is tip of the dissatisfaction iceberg because most American customers don't complain.
  - Along with long hold times, it can take longer to actually connect with someone who can answer your question. According to SSPA, the average time to get a response from a capable technician is 30 minutes for PC/Shrink-Wrap products.
  - At peak times, 85% of calls into tech support get busy signals.
  - 58% of support staff get less than 1 week of training before independently handling phone calls.
- Business' cost of ownership of a PC is often estimated at $8000 to $11,000 per year.

Supplemental Slides

When you talk about negligence in your class, the following slides might be helpful

Negligence Resulting in Damage or Injury

- **Negligence:** A vendor has a duty to make/design its products such that they do not create an unreasonable risk of injury or property damage. If the vendor breaches this duty, and that breach results in harm, the vendor is negligent and is liable to the victim for all losses resulting from the negligence.
- **Cost-of-Quality analysis** balances seller's prevention, appraisal and internal failure costs (including cost to repair) against the seller's external failure cost.
- **Negligence analysis** balances seller's prevention, appraisal, and internal failure costs (especially costs associated with repair) against society's external failure cost.

Comparison of Negligence and Contract

- **Negligence theories**: focus on public safety and public reliance on expert services. We look at the product *and* the process involved in creating the product.
- **Contract theories**: focus on living up to an agreement and on the quality and acceptability of the product or service as delivered. Process is irrelevant unless you contract to follow a specific process.

- (A *legal theory* is more like a template than a theory. It is a specification for a lawsuit. It lays out the conditions under which a lawsuit of a specified type is valid.)

Contracts vs Negligence

Contracts

- Law of quality
- Duty is to give the customer what s/he paid for.
- Likely types of suits:
  - corrupts or loses its own data
  - doesn't work; never delivered
  - erroneous reports
  - bugs that waste time or make the program hard to use
  - compatibility features don't work
  - cost-reduction promises aren't realized

Negligence

- Law of safety
- Duty is to make products that are not unreasonably unsafe.
- Likely types of suits:
  - corrupts or loses data obtained from some other program
  - damages connected peripherals
  - injures the user
  - injures customer who follows its directions
  - embedded software causes accidents
  - UI design causes accidents

Supplementary Slides

Tort Reform, Litigation Madness

Supplementary Background on Litigation

- There are a lot of myths about lawsuit-happy Americans and their evil plaintiffs' lawyers suing nice little companies like Enron into the ground.
- Speaking as an evil plaintiff's lawyer, I have a different impression.
- The following slides are not necessarily appropriate for your primary set of class notes. They are useful as a reserve. If you have a student who unreasonably argues that lawsuits aren't justified, we should see companies as victims, etc., then these notes are useful.
- (By the way, some of the problems were very real. Go back to the notes on legal tradeoffs and think about the Technological Approach to accountability allocation. That's the negligence approach. Applied too vigorously, it nearly killed the small aircraft industry and has had substantial unfair impacts on a few other industries.)

Quality-Related Litigation

- **Foundation of Quality-Related Litigation**: Quality-related litigation reflects the customer's intent to transfer its failure costs back to the company whose defective product caused them.
- If you don't estimate the extent of problems you're about to give your customers, you risk being blindsided by expensive litigation.
- Ford was caught off-guard when the courts (juries, judges, and appellate judges through the states' supreme courts) decided that its cost / benefit analysis was fundamentally unreasonable and charged it enormous punitive damages.
- Since then, several companies have been hit with large verdicts by juries and judges. I think it is fair to generalize that a company that sells a product that it knows is unsafe because of a defect, will face the wrath of the American legal system if someone is seriously injured or killed because of that defect.

Quality-Related Litigation

- In his classic book, “Out of the Crisis,” W.E. Deming named seven “Deadly Diseases” that afflict American businesses. Number 7 was “Excessive costs of liability, swelled by lawyers that work on contingency fees.” (p. 98)
  - *Are people who sue unreasonable?*
  - *Do lawsuits force companies to overspend on quality or safety?*
  - *How do we decide that a company overspends on quality or safety?*

Debunking Myths

- As long as we think of lawsuits as demonic threats to the public, we'll be too busy listening to our prejudices to understand the laws governing quality.
- So, let's do a little debunking . . . .
- Companies whose business practices lead them into situations of high risk for successful big-ticket lawsuits have an enormous interest in convincing the public that they should not be sued, that the laws should be changed, that the system is unfair.
- Some of those companies, and their favored politicians, make their cases with statistics. Others make their cases with "case studies".

Not-So Funny Statistics—A Litigation Crisis?

- Pete Wilson, then a candidate for the 1996 presidential election, made a big fuss about the following data. These allegedly proved that California (and the United States as a whole) was having a litigation crisis, which should be managed by laws making it harder for consumers to successfully sue companies.

- 1994 Annual Report of the Judicial Council of California

  Superior Court Civil Filings:

1983-84	1992-93	increase
561,916	684,070	122,154
		(21.7%)

- The Superior Court hears cases over $25,000. 1983-84 is the first of the 10 years in this study. 1992-93 is the last of the 10 years.

Not-So Funny Statistics—A Litigation Crisis?

But Look More Closely:

Superior Court Civil Filings:

1983-84	1992-93	*increase*	
561,916	684,070	*122,154*	*(22%)*

Personal injury, death, property damage:

1983-84	1992-93	*increase*	
96,731	88,346	*-8,385*	*(-9%)*

Other civil petitions (Primary increase is for child support):

1983-84	1992-93	*increase*	
121,968	267,980	*146,012*	*(120%)*

Not-So-Funny Stories – A Litigation Crisis?

- When I was in law school, the poster child of the lawsuits-are-unreasonable movement was a lawsuit filed by a criminal who was burglarizing a school. He fell through a skylight, injured himself and sued the school. To everyone's outrage, he won.
- Then we read the case. Here's how I remember it. The criminal was a student at that school. Every year before a big football game, some students would break into the school, go on the roof and post stuff (banners, whatever). They weren't supposed to do this, but it was a traditional (if illegal) prank. The school's janitor advised the principle of some cracks in the skylight windows on the roof. One student stepped on a cracked window, fell through, injured himself, and sued the school. Not surprisingly, he won. The school knew of a dangerous condition on the roof, and knew or should have known that people would be walking on the roof in the dark (unlikely to see the cracks), and didn't fix anything or post a warning, resulting in injury of one of its students.
- Cases like this came and went. A more recent poster-child was a lawsuit against McDonalds. Here are some plaintiff-side notes.

McDonalds and Spilled Coffee

(Courtesy of the Consumers Attorneys of California)

There is a lot of hype about the McDonald's scalding coffee case. No one is in favor of frivolous cases or outlandish results; however, it is important to understand some points that were not reported in most of the stories about the case. McDonald's coffee was not only hot, it was scalding capable of almost instantaneous destruction of skin, flesh and muscle. Here is the whole story.

Stella Liebeck of Albuquerque, New Mexico, was in the passenger seat of her grandson's car when she was severely burned by McDonald's coffee in February 1992. Liebeck, 79 at the time, ordered coffee that was served in a styrofoam cup at the drivethrough window of a local McDonald's.

After receiving the order, the grandson pulled his car forward and stopped momentarily so that Liebeck could add cream and sugar to her coffee. (Critics of civil justice, who have pounced on this case, often charge that Liebeck was driving the car or that the vehicle was in motion when she spilled the coffee; neither is true.) Liebeck placed the cup between her knees and attempted to remove the plastic lid from the cup. As she removed the lid, the entire contents of the cup spilled into her lap.

McDonalds and Spilled Coffee - 2

The sweatpants Liebeck was wearing absorbed the coffee and held it next to her skin. A vascular surgeon determined that Liebeck suffered full thickness burns (or third degree burns) over 6 percent of her body, including her inner thighs, perineum, buttocks, and genital and groin areas. She was hospitalized for eight days, during which time she underwent skin grafting. Liebeck, who also underwent debridement treatments, sought to settle her claim for $20,000, but McDonald's refused.

During discovery, McDonald's produced documents showing more than 700 claims by people burned by its coffee between 1982 and 1992. Some claims involved third degree burns substantially similar to Liebecks. This history documented McDonald's knowledge about the extent and nature of this hazard.

McDonald's also said during discovery that, based on a consultants advice, it held its coffee at between 180 and 190 degrees Fahrenheit to maintain optimum taste. He admitted that he had not evaluated the safety ramifications at this temperature. Other establishments sell coffee at substantially lower temperatures, and coffee served at home is generally 135 to 140 degrees.

McDonalds and Spilled Coffee - 3

Further, McDonald's quality assurance manager testified that the company actively enforces a requirement that coffee be held in the pot at 185 degrees, plus or minus five degrees. He also testified that a burn hazard exists with any food substance served at 140 degrees or above, and that McDonald's coffee, at the temperature at which it was poured into styrofoam cups, was not fit for consumption because it would burn the mouth and throat. The quality assurance manager admitted that burns would occur, but testified that McDonald's had no intention of reducing the "holding temperature" of its coffee.

Plaintiff's expert, a scholar in thermodynamics applied to human skin burns, testified that liquids, at 180 degrees, will cause a full thickness burn to human skin in two to seven seconds. Other testimony showed that as the temperature decreases toward 155 degrees, the extent of the burn relative to that temperature decreases exponentially. Thus, if Liebeck's spill had involved coffee at 155 degrees, the liquid would have cooled and given her time to avoid a serious burn.

McDonald's asserted that customers buy coffee on their way to work or home, intending to consume it there. However, the company's own research showed that customers intend to consume the coffee immediately while driving.

McDonalds and Spilled Coffee - 4

McDonald's also argued that consumers know coffee is hot and that its customers want it that way. The company admitted its customers were unaware that they could suffer third degree burns from the coffee and that a statement on the side of the cup was not a "warning" but a "reminder" since the location of the writing would not warn customers of the hazard.

The jury awarded Liebeck $200,000 in compensatory damages. This amount was reduced to $160,000 because the jury found Liebeck 20 percent at fault in the spill. The jury also awarded Liebeck $2.7 million in punitive damages, which equals about two days of McDonald's coffee sales.

Postverdict investigation found that the temperature of coffee at the local Albuquerque McDonald's had dropped to 158 degrees fahrenheit.

The trial court subsequently reduced the punitive award to $480,000 or three times compensatory damages even though the judge called McDonald's conduct reckless, callous and willful. Subsequent to remittitur, the parties entered a postverdict settlement.

A Complex Issue

- There is no doubt that several unreasonable lawsuits have been filed, especially for relatively small amounts. (It is often cheaper to pay off a claim for $10,000 than fight it.)
- Most of the unreasonable suits that get to court are tossed out, but it costs the defendant time and money to fight them even this far.
- A few of the unreasonable suits “score” – but only if they are persuasive. (When you read all the facts, they seem less unreasonable.)
- Most punitive damage awards are for less than $1000.
- If a jury awards large punitive damages, the judge and an appellate panel of judges, and often supreme court judges review the case and the award of damages. All of the reviews must conclude that the defendant’s conduct was so outrageous as to warrant punitive damages.

A Complex Issue

- But still, innocent companies pay too much to settle frivolous cases. They spend too much defending themselves. And occasionally, they lose.
- Is the law “bad” because it yields erroneous decisions sometimes?
- Unfortunately, we go back to the Allocation of Accountability tradeoffs. All of the accountability structures will result in some errors. We control the bias of the system (who should be at the greater risk of an error), but wherever you set the bias, someone will be unjustly treated.
- Certainly, we can work to make the system more accurate and more fair. But any system that uses classification rules that simplify (ignore relevant data associated with) the objects being classified, is bound to misclassify some of the time.
- So, who should we bias the system to favour or disfavour? Why?

Interesting Questions

The Licensing Question:

Should a manufacturer be able to restrict what you do with its software?

Should a manufacturer be able to restrict what you do with its software?

- Background:
  - Property law—doctrine of alienation. The seller gives up all rights in the sold property, the buyer can do what she wants with the property, including resale. Doctrine of alienation is considered basic public policy in property law—the sale alienates the property (real property or goods) from the seller.
  - Patent law—doctrine of exhaustion. Applies to goods that contain embedded patented technology. The buyer can do what she wants with the goods, including resell them. The buyer cannot create new goods of the same kind, because that would infringe the patent.
  - Copyright law—first sale doctrine. Applies to copies of copyrighted works (e.g. books, records, software – the work is the intellectual property, as contained in the original manuscript or sound recording. The copy is the published book that you buy in the store. You own that one individual book, not the rights to publish the book or make new copies). The buyer can do what she wants with the copies, including resell them. The buyer cannot create new copies, because that would infringe the copyright.

Should a manufacturer be able to restrict what you do with its software?

- Background:
  - The copyright and patent doctrines were decided early in 20th century, in response to attempts by book publishers and makers of some goods to restrict what people could do with their products or to limit their right to resell the products (used). The US Supreme Court rejected the restrictions and rejected the argument that a statement on the book or the product that said the item is sold "subject to a license" creates a "license" and an enforceable restriction.
  - There are no provisions for use restrictions in Article 2.
- We'll discuss use restrictions here, transfer restrictions in the next slide.
- Some references:
  - Brennan, Lorin (2000), Symposium on Approaching E-commerce Through Uniform Legislation: Understanding the Uniform Computer Information Transactions Act and the Uniform Electronic Transactions Act: Why Article 2 Cannot Apply To Software Transactions. *See brennan.rtf.*
  - Dodd, Jeff C.(1999), Time And Assent In The Formation Of Information Contracts: The Mischief Of Applying Article 2 To Information Contracts, *See dodd.rtf.*
  - Gomulkiewicz, Robert W. (1998), The License Is The Product: Comments on the Promise of Article 2B for Software and Information Licensing. *See gomulkiewicz.rtf*
  - Nimmer, David , Elliot Brown, & Gary N. Frischling (1999) The Metamorphosis of Contract into Expand. *See david_nimmer.rtf*

Should a manufacturer be able to restrict what you do with its software?

- UCITA use restrictions.
  - The problem / opportunity faced by the software industry is that to use a piece of software, you have to make a copy of it. This isn't true for books or patented machines.
  - The UCITA authors argue that this makes software "sales" fundamentally different from other sales of goods, and they propose a licensing model.
  - Background note: According to Watts Humphrey, the co-author of the first software license, licensing was adopted as a model of choice for software transactions by IBM, when the Copyright Office and the Patent & Trademarks Office were refusing to register copyrights and patents of software. The only avenue left for intellectual property protection at that time was licensing. This defensive strategy grew into a much broader approach to controlling customer use and limiting vendor liability, far beyond IBM's original intent.

Should a manufacturer be able to restrict what you do with its software?

UCITA use restrictions.

- UCITA section 102(a) (19) defines “Contractual use term” as “an enforceable term that defines or limits the use, disclosure of, or access to licensed information or informational rights, including a term that defines the scope of a license.”
- Under UCITA section 102(a)(57), a scope term defines “(A) the licensed copies, information, or informational rights involved; (B) the use or access authorized, prohibited, or controlled; (C) the geographic area, market, or location; or (D) the duration of the license.”
- UCITA section 307 (b) states that “If a license expressly limits use of the information or informational rights, use in any other manner is a breach of contract.”

Definitions that you might need from UCITA

Section 102(a)

- (9) "Computer" means an electronic device that accepts information in digital or similar form and manipulates it for a result based on a sequence of instructions.
- (10) "Computer information" means information in electronic form which is obtained from or through the use of a computer or which is in a form capable of being processed by a computer. The term includes a copy of the information and any documentation or packaging associated with the copy.
- (11) "Computer information transaction" means an agreement or the performance of it to create, modify, transfer, or license computer information or informational rights in computer information. The term includes a support contract under Section 612. The term does not include a transaction merely because the parties' agreement provides that their communications about the transaction will be in the form of computer information.
- (12) "Computer program" means a set of statements or instructions to be used directly or indirectly in a computer to bring about a certain result. The term does not include separately identifiable informational content.
- (35) "Information" means data, text, images, sounds, mask works, or computer programs, including collections and compilations of them.
- (36) "Information processing system" means an electronic system for creating, generating, sending, receiving, storing, displaying, or processing information.
- (37) "Informational content" means information that is intended to be communicated to or perceived by an individual in the ordinary use of the information, or the equivalent of that information.
- (38) "Informational rights" include all rights in information created under laws governing patents, copyrights, mask works, trade secrets, trademarks, publicity rights, or any other law that gives a person, independently of contract, a right to control or preclude another person's use of or access to the information on the basis of the rights holder's interest in the information.

Should a manufacturer be able to restrict what you do with its software?

- Questions for discussion
  - What's being restricted here?
  - What kinds of things are subject to the restrictions? (Computer programs. Anything else?)
  - What if these restrictions applied to the software sold inside a car?
  - Are these restrictions appropriate for:
    - Commercial software?
    - Consumer software?
    - Life-critical software?
    - Electronic books?
    - Computers or peripherals?

Interesting Questions

The licensing question: Should a manufacturer be able to restrict your transfer of its software?

Should a manufacturer be able to restrict your transfer of its software?

- Read the notes on licensing in the last section.

UCITA 102(a) (65) "Transfer":

- (A) with respect to a contractual interest, includes an assignment of the contract, but does not include an agreement merely to perform a contractual obligation or to exercise contractual rights through a delegate or sublicensee; and
- (B) with respect to computer information, includes a sale, license, or lease of a copy of the computer information and a license or assignment of informational rights in computer information.

SECTION 502. TITLE TO COPY.SECTION 502. TITLE TO COPY.

- (a) In a license:
- (1) title to a copy is determined by the license;
- (2) a licensee's right under the license to possession or control of a copy is governed by the license and does not depend solely on title to the copy; and
- (3) if a licensor reserves title to a copy, the licensor retains title to that copy and any copies made of it, unless the license grants the licensee a right to make and sell copies to others, in which case the reservation of title applies only to copies delivered to the licensee by the licensor.

Should a manufacturer be able to restrict your transfer of its software?

- Under Section 502, the publisher of software (or other electronic information) can restrict your ability to transfer a copy of the software even if you own that copy (like you own a book).
- The vendor can specify that you may not lend, sell or give away your copy of the program. (Remember, when you transfer the copy, you take it OFF your computer. We're not talking about piracy.)
- This restriction kills the used software market.
- The American Intellectual Property Law Association (the primary intellectual property lawyers professional society) and the intellectual property sections of New York City's Bar Association (arguably the sharpest collection of copyright and entertainment lawyers in the world) both opposed UCITA, arguing that this provision repeals the First Sale doctrine.
- NOTE: Many intellectual property lawyers believe that these restrictions will be preempted by the Copyright Act. If there is preemption, the rules will probably depend on whether the software is custom or mass-market.

Should a manufacturer be able to restrict your transfer of its software?

- NOTE: Now that UCITA is failing (opposition strength will probably block it from future enactments in any state), the UCITA policies are being pushed through other channels. For example, the Intellectual Property Law section of the American Bar Association has formed a committee on transfers of electronic information. The same rules will be advocated in that committee (with or without success).
- QUESTIONS:
  - Why should you be able to transfer books but not software?
  - Why should you be able to transfer printed books but not electronic copies of the same books?
  - What risks (to vendors, sellers, and society) are posed by a market in used software?

Interesting Questions

Is licensing an appropriate model for off-the-shelf (especially mass-market) software?

Is licensing an appropriate model for off-the-shelf (especially mass-market) software?

- UCITA distinguishes between consumer software, mass-market software, and the rest.
  - Consumer software is finished product for household use--non-business, non-professional. Computer games are consumer software, but using a word processor at home to catch up on work from the office (or to create course notes if you're a teacher) is not.
  - Mass-market products are finished product sold in consumer quantity (typically one copy), in a consumer marketplace, are not customized, are not access contracts (e.g. use of AOL is not "mass-market) unless they are consumer uses, and are sold to consumers or businesses.
  - "Everything else" includes free software, beta software, COTS software for narrow markets (e.g. dentist office accounting), high end custom software, access contracts to data online, etc.

Is licensing an appropriate model for off-the-shelf (especially mass-market) software?

- Let's ignore the details of the distinctions made in UCITA and think about distinctions in principle.
  - Under what circumstances are *transfer restrictions* absolutely reasonable? Are there circumstances under which they would be unreasonable?
  - Under what circumstances are *restrictions on who can use the product or how many people can use it (one at a time)* absolutely reasonable? Are there circumstances under which they would be unreasonable?
  - Under what circumstances are *restrictions on disclosure or other speech* absolutely reasonable? Are there circumstances under which they would be unreasonable?
  - Under what circumstances are *warranty limitations* absolutely reasonable? Are there circumstances under which they would be unreasonable?
  - Under what circumstances are *remedy limitations* absolutely reasonable? Are there circumstances under which they would be unreasonable?

Interesting Questions

What limits should a manufacturer be able to place on what you (customer or journalist) can say about its software?

What limits should a manufacturer be able to place on what you (customer or journalist) can say about its software?

- UCITA section 102(a) (19) defines “Contractual use term” as “an enforceable term that defines or limits the use, disclosure of, or access to licensed information or informational rights, including a term that defines the scope of a license.”
- UCITA section 307 (b) states that “If a license expressly limits use of the information or informational rights, use in any other manner is a breach of contract.”
- The plain language of the statute is that the software vendor has the right to forbid customers from writing about the software. And in fact, some vendors (such as Microsoft, Oracle, and Network Associates) do include such restrictions, forbidding people from publishing the results of benchmark studies or other product comparisons and/or forbidding people from publishing product reviews, or from using the product to create or public criticisms of the vendor.

What limits should a manufacturer be able to place on what you (customer or journalist) can say about its software?

- The Official Comments in UCITA make it sound as though such restrictions would not be enforced or are not approved. Unfortunately, the "Comments" are not part of the UCITA law. The state legislatures adopt a statute without adopting the NCCUSL comments, so you (and the judge) have to go by what the statute says, not what its authors say it says.
- One of the complications in interpreting UCITA is that, over the years, the UCITA drafting process became polarized and bitter. It's impossible to tell what the authors of UCITA really believed they were saying, but here, as in several other places, UCITA critics offered alternative statutory language that would make UCITA less ambiguous and would tie the UCITA comments more closely to the UCITA language. Here, as elsewhere, those suggested changes were rejected. The American Bar Association Task Force's primary criticism of UCITA was that they found it "inscrutable" – after extended study, they were often unable to determine what the actual legal rule was.
  - American Bar Association Working Group Report On The Uniform Computer Information Transactions Act, See aba_ucita.pdf
- The UCITA drafting committee finally did propose an amendment to UCITA that appeared to make progress, but critics pointed to what they considered to be truck-sized loopholes.

What limits should a manufacturer be able to place on what you (customer or journalist) can say about its software?

- Rather than argue about what UCITA itself might or might not say, let's face the fact that some software companies include restrictions on speech in their license agreements. Some references:
  - Foster, Ed, John Fontana, John, and Steve Murchie, Articles (and MS Response) on Microsoft's refusal to allow publication of benchmarks. See microsoft_forbids_benchmarks.rtf
  - McAfee has also included speech restrictions with Viruscan.

    "The customer shall not disclose the results of any benchmark test to any third party without McAfee's prior written approval."

    "The customer will not publish reviews of the product without prior consent from McAfee."
  - The Attorney-General of New York is now prosecuting Network Associates (McAfee) on the theory that this is a deceptive trade practice. For details, see
    - People of the State of New York v. Network Associates (2002), Memorandum of Law in Support of Petition of Attorney-General, see nyntwrkass020702mol.pdf and Petition, see nyntwrkass020702pet.pdf.

What limits should a manufacturer be able to place on what you (customer or journalist) can say about its software?

Questions:

- Is it ever fair for a vendor to restrict customers' speech about products?
  - What about beta software, given to customers under nondisclosure agreements?
  - What about leading edge custom software that really does contain critical trade secrets?
- Microsoft says that they place these restrictions in order to protect themselves from misrepresentation. Network Associates used to say essentially the same thing. *Should they be allowed to be the judges, before publication?* Why or why not?
- A free market economy involves extensive freedom of contract—your ability to enter into whatever deal you want to enter into, without government interference. How can you make intelligent choices if vendors can suppress negative reports?
  - For more on speech, see Dreyfuss, Rochelle Cooper (1999), Do You Want to Know a Trade Secret? How Article 2B Will Make Licensing Trade Secrets Easier (But Innovation More Difficult). *See dreyfuss.rtf*.

Interesting Questions

What limits should a manufacturer be able to place on your ability to reverse engineer or study its software?

What limits should a manufacturer be able to place on your ability to reverse engineer or study its software?

- "A common criticism of software publishers is that their EULAs prohibit reverse engineering, decompilation, and disassembly of their software. Software publishers typically restrict these activities because they risk exposing, and hence losing, to the public domain, the publisher's crown jewel-the secrets contained in the software's source code. Most purchasers of off-the-shelf software, however, care little, if at all, about the right to reverse engineer, and they certainly are not interested in paying more money to acquire this right. The entities that are most interested in acquiring this right are competitors of the software developer. ***A competitor should not be permitted to acquire the right to examine a company's trade secrets for the low price that the typical end user pays for the software."***
- --- Robert W. Gomulkiewicz (a senior corporate attorney at Microsoft) and Mary L. Williamson (1996)
- (A EULA is an end user license agreement, typically typically shrink-wrapped and presented to the customer after the sale.)

What limits should a manufacturer be able to place on your ability to reverse engineer or study its software?

- Anything that makes it more expensive for newcomers to develop or market competitive products serves the interests of entrenched market leaders.
- Reverse engineering restrictions have been upheld in situations involving negotiated contracts,[1] but not mass-market contracts. The case of Sony v. Connectix[2] is the latest in a line of cases rejecting the idea that reverse engineering of a product sold (or licensed) in the mass-market is copyright infringement.[3] The Digital Millennium Copyright Act[4] expressly permits reverse engineering for the purpose of creating interoperable products, but it is silent about reverse engineering done to learn unprotectable ideas embedded in software for the purpose of developing a competing product.

What limits should a manufacturer be able to place on your ability to reverse engineer or study its software?

- A recent case under the DMCA (Universal Studios v. Corley, the DeCSS case [9]) appears to restrict the right to reverse engineer. But the ruling judge made it clear that this wasn't his intent. The defendant had reverse engineered, successfully broken an encryption scheme that controlled access to digital content (breaking encryption to defeat access control is a specific DMCA no-no), and then published the means to decrypt (a DMCA NO-NO). A person is allowed to reverse engineer to attempt to make two products interoperable. The defense argued that the purpose of the exercise was to make Linux-based computers able to read DVDs.
- Judge Kaplan didn't believe that the defendant actually did the decryption with the goal of interoperability with Linux, and so concluded that the DMCA's exception (circumvention of access-control/copy-prevention mechanisms is allowed IF it is done for interoperability) did not apply.
- Additionally, the defendant didn't reverse engineer and then simply publish a Linux driver or sell a DVD player suitable for Linux machines. Instead, he published the decryption scheme, which could be used for much broader purposes (piracy) than just reading a DVD on Linux machines.

What limits should a manufacturer be able to place on your ability to reverse engineer or study its software?

- Many people are concerned about the DeCSS case because it is the first one that held a defendant liable for reverse-engineering a mass-market product. Against the strong tradition of respect for reverse engineering in the courts, it remains to be seen whether this case will extend much beyond its specific facts.
- UCITA sections 102(a)(19) and 102(a)(57) allow the publisher to define or limit the use of licensed information (including software).[6] A ban on reverse engineering is just another use restriction. UCITA section 307(b) states that "If a license expressly limits use of the information or informational rights, use in any other manner is a breach of contract." On the face of the statute, any reverse engineering done in violation of a no-reverse-engineering use restriction is a breach of contract.

What limits should a manufacturer be able to place on your ability to reverse engineer or study its software?

- Given that this is a good restriction under UCITA, a mass-market publisher's restriction on reverse engineering will be upheld unless a court determines that it is preempted by federal law, or in violation of a fundamental public policy,[7] or unconscionable.[8] I think that it is likely that the courts will eventually find that license terms that ban reverse engineering in a mass market are still preempted by the fair use doctrine[9] but it will probably take years of litigation to establish this. Until then, the language of UCITA will have a chilling effect on law-abiding engineers.
- Remember that reverse engineering is entirely lawful and respectable under American law. All other industries have to live with it. Manufacturers can protect their inventions by patenting them. You can't work around the protection of a patent through reverse engineering.
- This is another example of a fair use issue that was raised repeatedly over the years, in the drafting committee meetings and in public debate and that was addressed in the McManis and Perlman resolutions, passed respectively by the ALI and NCCUSL, and in various other suggestions.

What limits should a manufacturer be able to place on your ability to reverse engineer or study its software?

[1]. See Raymond T. Nimmer, The Law of Computer Technology (Rev. Ed., including Supp. 3, 1999).

[2]. Sony Computer Entertainment, Inc. v. Connectix Corp., 2000 U.S. App. LEXIS 1744 (9th Cir. 2000).

See also Karas, Stan (2001) INTELLECTUAL PROPERTY A. COPYRIGHT 2. DEFENSES A) FAIR USE: Sony Computer Entertainment, Inc. v. Connectix Corp. *see karas.rtf*

[3]. For example, Sega Enterprises Ltd. v. Accolade, Inc., 977 F.2d 1510 (9th Cir. 1992); Vault Corp. v. Quaid Software Ltd., 847 F.2d 255 (5th Cir. 1988).

[4]. 17 U.S.C. § 1201(f). (Digital Millenium Copyright Act)

Reverse Engineering. –

(1) Notwithstanding the provisions of subsection (a)(1)(A), a person who has lawfully obtained the right to use a copy of a computer program may circumvent a technological measure that effectively controls access to a particular portion of that program for the sole purpose of identifying and analyzing those elements of the program that are necessary to achieve interoperability of an independently created computer program with other programs, and that have not previously been readily available to the person engaging in the circumvention, to the extent any such acts of identification and analysis do not constitute infringement under this title.

(2) Notwithstanding the provisions of subsections (a)(2) and (b), a person may develop and employ technological means to circumvent a technological measure, or to circumvent protection afforded by a technological measure, in order to enable the identification and analysis under paragraph (1), or for the purpose of enabling interoperability of an independently created computer program with other programs, if such means are necessary to achieve such interoperability, to the extent that doing so does not constitute infringement under this title.

(3) The information acquired through the acts permitted under paragraph (1), and the means permitted under paragraph (2), may be made available to others if the person referred to in paragraph (1) or (2), as the case may be, provides such information or means solely for the purpose of enabling interoperability of an independently created computer program with other programs, and to the extent that doing so does not constitute infringement under this title or violate applicable law other than this section.

(4) For purposes of this subsection, the term ‘‘interoperability’’ means the ability of computer programs to exchange information, and of such programs mutually to use the information which has been exchanged.

What limits should a manufacturer be able to place on your ability to reverse engineer or study its software?

[5]. Universal Studios v. Corley (1/29/02), United States Court of Appeals for the Second Circuit, *see Universal2002.rtf*

Universal Studios v. Remerdes (8/17/2000) United States District Court for the Southern District of New York, *see Universal0800.rtf*

[6]. In its zeal to redefine the language, UCITA defines computer programs as information. See UCITA § 102(a)(35). "Information" means data, text, images, sounds, mask works, or computer programs, including collections and compilations of them." Id.

[7]. UCITA § 105(b).

[8]. Id. § 111.

[9]. Vault Corp. v. Quaid Software Ltd., 847 F.2d 255, explicitly rejected a software license's ban on reverse engineering in a mass-market license.

See also

- Georgenson, Karen (1996), Reverse Engineering of Copyrighted Software: Fair Use Or Misuse? See georgenson.rtf.
- Lewis, Terril (2000), Reverse Engineering Of Software: An Assessment Of The Legality Of Intermediate Copying. *See lewis.rtf.*

What limits should a manufacturer be able to place on your ability to reverse engineer or study its software?

- Perlman amendment (passed by NCCUSL)
  - "If a court as a matter of law finds the contract or any term of the contract to have been unconscionable or contrary to public policies relating to innovation, competition, and free expression at the time it was made, the court may refuse to enforce the contract or it may enforce the remainder of the contract without the impermissible term as to avoid any unconscionable or otherwise impermissible result."
- Corresponding UCITA change
  - "If a contract term violates a **fundamental** public policy, the court may refuse to enforce [it]. . . **to the extent that** the interest in enforcement **is clearly outweighed by** a public policy against enforcement of that term."
- The December 2001 proposed amendments from the UCITA authors permit some reverse engineering to support interoperability only. Software vendors can still ban reverse engineering for all other purposes, including security analysis (for which DMCA permits reverse engineering).

What limits should a manufacturer be able to place on your ability to reverse engineer or study its software?

- Here's why colleagues and I have reverse engineered:
  - Personal education.
  - Understand and work around (or fix) limitations and defects in tools that I was using.
  - Understand and work around (or fix) defects in third-party products.
  - Discover security flaws in software and work around (or fix) them.
  - Make my product compatible with (able to work with) another product.
  - Make my product compatible with (able to share data with) another product.
  - To learn the principles that guided a competitor's design.
  - Determine whether another company had stolen and reused some of my company's source code.
  - Determine whether a product is capable of living up to advertised claims.
    - *See Kaner, Cem (1998), The Problem of Reverse Engineering, kaner_revengr.rtf.*

Interesting Questions

What limits should a manufacturer be able to place on your ability to compete with its software?

What limits should a manufacturer be able to place on your ability to compete with its software?

- Question: How can an established vendor keep out competitors without engaging directly in anticompetitive practices?
- Here are some possible thoughts:
  - Ban independent magazine reviews (or at least, reviews that are based on actual experience with your product or service). That way, the consumer will choose primarily on the basis of advertising—and you're bigger, so you can spend more on advertising.
  - Don't publish contract terms until after the sale. This makes it hard for journalists and customers to compare terms. If a competitor offers a better deal on warranty or support, so what? Who will know?
  - Directly ban, in your use restrictions, use of your product to create products that will compete with you.
  - Ban reverse engineering, making it harder for competitors to make products that work with yours, or that work like yours, or to prove that yours doesn't work the way (or as well as) you claim.
  - Extend the doctrine of implied warranties to software that is given away, increasing the risks and costs of the free and open source models of competition with the closed source model.

What limits should a manufacturer be able to place on your ability to compete with its software?

- What else do you think you'd need?
- What else do you think you could get under a freedom-of-contract regime like UCITA's?
- Question: Why do you think the largest software companies support UCITA?
- ==========
- Another interesting discussion of licensing and competition, not focused on UCITA, is
  - Carroll, Darren J. (1992) When More Is Less: Controlling The Market For Computer Software Enhancements. See carroll.rtf.

Interesting Questions

What measures should a software vendor be able to take in the face of unauthorized use or use beyond the license?

What measures should a software vendor be able to take in the face of unauthorized use or use beyond the license?

- I'm not going to try to approach this with any comprehensiveness here, but there is plenty of discussion on the Net.
- Suppose you're a small company, and you license your software to a big company. You give them a 100-seat license and they use it at 200 machines. You ask them (demand) to either pay for an extended license or quit using it on so many machines. They thumb their lawyers' noses at you.
  - Should you have the right to shut down their use of your system? Should you be able to do this on your own, or should a judge be involved?
  - What other remedies do you think you have, beyond shutting them down immediately?
  - What if you're in the USA and they're in some far-distant country?

What measures should a software vendor be able to take in the face of unauthorized use or use beyond the license?

- Suppose you publish game software, under different licenses:
  - Trial version – 3 days free, then it expires.
  - Complete play version – play to the end or for 3 months, then it expires.
  - Number of boots version – boot the program more than 100 times and it expires.
  - One machine only version – play it as much as you want on one particular computer, but it self-destructs rather than moving to a new machine.
- Should you be able to license or sell the software under terms like these? Why or why not?
- If you say, *NOT,* then what do you think of letting people rent cars? Would it be fair for a car-renter to give you the car for three days and set it to automatically shut down if it's gone more than 4 days?

Computer Tampering, Conversion

- In Clayton X-Ray Co. v. Professional Systems Corp. (WD 43583, Mo. Ct. App., W.D. 8/6/91; 9 Computer Law 38). PSC introduced a time bomb b/c Clayton had not fully paid for the software. The bomb shut down the system. PSC refused to turn the system back on until Clayton paid for it. Jury awarded punitive and compensatory damages.
- This is not a case of time bomb in the initial product. This is a time bomb introduced after the fact, by intrusion into plaintiff's computer.
- Cases like this have made many vendors cautious about using "self-help" – shutting down customers' software or systems without a court order.

Interesting Questions

How should contracts in our industry be formed?

How should contracts in our industry be formed?

- UCITA validates almost all of the terms of shrink-wrap / click-wrap contracts.
- These are still controversial and courts are split on the enforceability of their harsher terms. For example, over the last few months:
  - Washington state Supreme Court *(M.A. Mortenson Co. v. Timberline Software)* approved a click-wrap disclaimer of warranties and remedy limitation, barring the plaintiff from recovering significant losses caused by a defect that was apparently known by the vendor but not disclosed to the customer. *See mortenson.rtf*
  - Federal court in Kansas (*Klocek v. Gateway, Inc.*) just rejected a shrink-wrapped arbitration clause, ruling that this was a proposed modification to the contract. *See klocek.rtf*

Step-Saver Data Systems

- Step-Saver (a vertical market reseller) repeatedly bought Multilink Advanced, an allegedly MS-DOS compatible operating system, from The Software Link (TSL). On each box was a disclaimer: the software was sold AS IS, without warranty; TSL disclaimed all express and implied warranties; and a purchaser who didn't agree to this disclaimer should return the product, unopened, to TSL for a refund.
- Step-Saver sued TSL, claiming that Multilink Advanced was not MS-DOS compatible. TSL argued that Step-Saver had accepted the terms of the warranty disclaimer when it opened each package, and therefore Step-Saver could not sue.
- Similar case: Arizona Retail Systems v. The Software Link (1993) United States District Court For The District Of Arizona. *See arizona_retail.rtf*

Should this disclaimer be valid for the first sale to Step-Saver?

Subsequent sales?

How should contracts in our industry be formed?

- These are controversial for several reasons:
  - The customer doesn't get to see the terms until after the sale. (That is, until she pays her money, takes delivery, and tries to install the software). *This is intentional—lawyers for software vendors actively resisted proposals that would require them to show terms to people who asked for them or that would require them to post terms on their website, or that would require them to provide a link, when someone is downloading software, that lets them see the terms before they pay.* Because the customer won't see the terms until after the sale, harsh terms won't kill the deal. The lawyers can get their great terms while the salesman still gets the sale.
  - See Braucher, Jean (2000), Delayed Disclosure in Consumer E-commerce as an Unfair And Deceptive Practice. See *braucher.rtf.*
  - There is no room for negotiation. There is no possibility of conflicting forms or conflicting understandings. The only choice the customer has is to use the software under the vendor's terms or not.

How should contracts in our industry be formed?

- These are controversial for several reasons:
  - The vendor's terms are *always* the contract terms. In olden days (before 1940's), vendors and customers would play a "battle of the forms" game. The idea was that the final set of terms received before the sale became a legal sale, became the contract that governed the sale. So, each side jockeyed to get their terms last. Article 2 of the Uniform Commercial Code stopped this game. Instead, when seller and buyer traded forms and neither explicitly agreed to the other's form, the agreement was put together piecemeal. There were some terms (like price) that the two parties actually agreed on. These were the foundation of the contract. There were other terms, especially minor terms, that neither side had protested and that didn't go to the heart of the deal. These were in the contract. There were also terms that conflicted—the vendor said one thing, the customer said another. In these cases, both terms were bounced and the judge would apply "default rules" supplied with the UCC. The intention was a scrupulous neutrality in the law as between buyer and seller. In commercial cases, where there is a negotiated contract, UCITA actually gives the shrinkwrap terms precedence over the terms of the negotiated deal. This is reinstatement of the battle of forms with a vengeance. It is stunning to see this come back, when the last holdout jurisdiction (the last courts to allow the final form to control the deal) finally rejected that approach and adopted *Step-Saver's* reasoning and rule as the mainstream U.S. approach. *See ionics.rtf.*

How should contracts in our industry be formed?

- These are controversial for several reasons:
  - The Step-Saver case applied a standard analysis when there is only one form contract but the customer doesn't get access to it until after the sale. In that case, we apply the terms that the customer and seller explicitly agreed to, apply the non-material terms that the customer didn't protest (didn't know about), and take the UCC default rules as the customer's form. Thus, if the seller disclaims warranties or eliminates remedies, but does so after the fact, these terms fail. The seller has to make its intention to adopt harsh terms known to the customer before the sale, or at least make it easy for the customer to find out those terms.
  - UCITA explicitly overrules Step-Saver. The seller always wins. The click OK when you install the software is treated as equivalent to a signature that says you agree to each and every term in this license that comes with this software.

How should contracts in our industry be formed?

- Up to 1995, there was great doubt that any court would ever uphold shrink-wrap or click-wrap licenses. Since 1995 (largely because of the ceaseless and prestigious advocacy of the UCITA drafting and cheerleading community), most cases have accepted the validity of click-wrap contracts, but several have rejected specific terms as overreaching or inconspicuous or unfairly surprising. The trend, as UCITA gained initial success, has been to accept more and more of the contract terms as if they had been fully negotiated and individually signed.
- The most recent case on these lines is I.LAN SYSTEMS V. NETSCOUT SERVICE LEVEL Corp. (2002), United States District Court For The District Of Massachusetts. *See ilan.rtf*

How should contracts in our industry be formed?

- Question:
  - If you were the vendor and you could (a) delay your terms until no one would read them and (b) enforce them anyway,
    - What terms would you put in your contract? Why?
    - Bonus: extra points for creativity.

Some Interesting Questions

- The licensing question: Should a manufacturer be able to restrict what you do with its software?
- The licensing question: Should a manufacturer be able to restrict your transfer of its software?
- Is licensing an appropriate model for off-the-shelf (especially mass-market) software?
- What limits should a manufacturer be able to place on what you (customer or journalist) can say about its software?
- What limits should a manufacturer be able to place on your ability to reverse engineer or study its software?
- What limits should a manufacturer be able to place on your ability to compete with its software?
- What measures should a software vendor be able to take in the face of unauthorized use or use beyond the license?
- How should contracts in our industry be formed?
- How should warranties be created and what should they cover?

- What standard of performance should we require of software service providers?
- What standard of performance should we expect of COTS software?
- What remedies should be available for business losses caused by defective software?
- What is a “consumer” of computer software? Who should be entitled to consumer protection?
- Should consumer protection laws developed for traditional goods apply to software too?
- Should there be a standard of unfairness for contract terms?
- Should embedded and non-embedded software be treated differently in the law?
- How should we distinguish between permissible and impermissible claims about a product?
- To what extent should we rely on electronic signatures?
- How can software engineers be involved in the legislative and judicial processes?

Interesting Questions

How should warranties be created for software and what should they cover?

How should warranties be created for software and what should they cover?

- There is extensive discussion of this in the materials (see e.g. my Software Engineering & UCITA paper, kaner_sweucita.rtf and in *Bad Software*.
- Highlights
  - Article 2 creates express warranties, based on what the vendor says about the product or on what the vendor shows (in demonstrations) about the product.
  - Statements made after the sale can be binding. See Kaner, Liability for Defective Documentation, *defective_doc.rtf*.
  - UCITA relaxes the Article 2 rule, for example, exempting user interface characteristics from warranty-by-demonstration.
  - Article 2 creates implied warranties (fitness, merchantability) and makes them hard to disclaim. Disclaimers must be genuinely conspicuous. UCITA relaxes the rule, allowing first revelation of a binding disclaimer to appear in the post-sale clickwrap.

How should warranties be created for software and what should they cover?

- A key argument behind this is that software is likely to be defective, and so strong warranty protection for customers will kill the industry. This was persuasive in the UCITA meetings and variants of it are published in the literature.
- Selman, Jeffrey C. & Christopher S. Chen (1997), Steering the Titanic Clear of the Iceberg: Saving the Sale of Software from the Perils of Warranties. See selman.rtf
- What guarantees do you think
  - Software publishers
  - Custom software developers
  - Software consultants
  - Software resellers
  - Free software developers / distributors

  should make to customers / users? Why? Should the structure (nature and content of the guarantees) be the same across the groups? Why?

Interesting Questions

What rules should we apply to innocent or negligent misrepresentation?

Misrepresentation

- False representation by the seller
- of a material (important) fact
- that the plaintiff justifiably relies on
- and as a result, the plaintiff is damaged.

- Misrepresentation can be:
  - Innocent
  - Negligent
  - Fraudulent
- A misrepresentation is fraudulent if the maker
  - knows or believes that the matter is not as he represents it to be, or
  - does not have the confidence in the accuracy of his representation that he states or implies, or
  - knows that he does not have the basis for his representation that he states or implies
- All states allow you to sue for fraud (and recover punitive damages). A few states allow suits for non-fraudulent misrepresentation (punitives probably unavailable).

Misrepresentation

Negligent misrepresentation

- The *duty* is to exercise the care or competence of a reasonable person who is communicating information.
- Not all misrepresenters will be held liable. Many states require a special relationship between the victim and misrepresenter, such as a position of trust.
- States vary in the degree to which they allow a negligent misrepresentation suit, in the face of an integration clause and no misrepresentation in the body of the contract.

Post-sale misrepresentation

- Post-sale fraud is actionable if it causes a person to forego or refrain from asserting an existing right or to change position in some other way.
- Several support staff are trained to deny knowledge of known bugs and some are told that they are supposed to lie if necessary to keep a customer from returning a product.
- *Ritchie Enterprises v. Honeywell Bull, Inc.* (1990) is a classic case in which the contract successfully disclaimed warranties, limited damages, etc., but the plaintiff was allowed to sue for post-sale misrepresentations (that problems would be or were being fixed).

Burroughs v. Hall Affiliates

- Hall imports artificial flowers, and bought a Burroughs B80-40 computer in 1977 to handle its accounting and inventory functions. The system didn't work and Hall sued claiming that Burrough's salespeoples' representations about the system were fraudulent. The court listed 4 representations:
  1. the machine would do inventory and accounting simultaneously
  2. the machine was capable of multiprogramming
  3. the machine was capable of operating a terminal display unit in a data communications environment
  4. the machine and all of its component parts were new.
- The court found that there was no evidence of intentional misrepresentation.

Should Burroughs be held liable for its mistake?

Winter v. G.P. Putnam

- Winter became seriously ill from picking and eating mushrooms after relying on The Encyclopedia of Mushrooms, published by Putnam. Putnam did not verify the material in the book and did not intentionally include the error.

  Should Putnam be held liable for this unintended misinformation?

  For more info, see Kaner, Liability for Defective Content, *kaner_content.rtf*.

Interesting Questions

What standard of performance should we require of software service providers?

Interesting Questions

What standard of performance should we expect of COTS software?

Interesting Questions

What about software that is unusable but conforms to the contract?

Family Drug Store

- The plaintiffs are a couple of pharmacists who bought a computer program known as the Medical Supply System from Gulf States. After they realized what they had bought, they asked for, and then sued for, a refund. Here were some of the problems of the system:
  (1) all data had to be printed out, and could not be viewed on the monitor;
  (2) the information on the monitor would appear in code;
  (3) numerical codes were needed in order to open a new patient file
  (4) the system was unable to scroll.
- The court found that the seller had not in any ay misrepresented the system, and that it as not useless even though it was awkward to use. Further, the price of the software was about $2500 compared to $10,000 for other packages.
- Should the plaintiffs get a refund?

Interesting Questions

What remedies should be available for business losses caused by defective software?

Remedies

- Under Article 2 and UCITA, a vendor can limit the remedies available to the customer, in the case of breach of contract.
- UCITA is a little more charitable to the vendor:
  - The vendor can make the contract non-cancellable: no cancellation even if there is a material breach.
  - The UCITA vendor can limit remedies inconspicuously, after the sale. Some states allow this under Article 2, some allow it only in business-to-business deals and some don't allow it (the vendor has to reveal the term pre-sale).
  - The UCITA vendor can modify the contract any time. If the contract *says* the vendor can modify it any time, and says that the customer can't cancel in the event of modification, the customer will be stuck in the contract. What used to be a remedy under the contract can be modified away.
  - The Article 2 vendor used to be responsible for "a minimum adequate remedy" if the expected remedies failed. UCITA has dropped this.
- QUESTION: If the vendor breaches the contract and ships defective product that causes losses, what should the vendor be liable for?

Interesting Questions

What is a “consumer” of computer software? Who should be entitled to consumer protection?

Interesting Questions

Should consumer protection laws developed for traditional goods apply to software too?

Should consumer protection laws developed for traditional goods apply to software too?

- UCITA does an interesting dance. The statute very explicitly says that when a consumer protection statute would apply to a consumer transaction, that statute's rules trump UCITA's.
- But UCITA defines transactions in information as licensing transactions. These are the first consumer licenses that lawmakers have had to face. So, consumer protection statutes talk about:
  - Sales (not licenses) of consumer goods (software is not a "good" under UCITA)
  - Sales or leases of goods and services (software is neither, and the transaction is not a sale or a lease)
- As a result, UCITA software has been pulled outside the scope of most consumer protection law.
- State legislatures can choose (of course) to amend their consumer protection statutes to include software. (Just imagine how many big-spending consumer protection advocates [most of whom make about $25,000 per year] are giving big donations to legislators to make sure they make all those amendments. Undoubtedly, they are a more powerful lobby than the big software companies, who would encourage legislators to be more choosy about which statutes to amend, right?)

Magnuson-Moss Act and Consumer Statutes

- Magnuson-Moss Warranty Improvement Act (Mag-Moss)
  - Passed in response to widespread abuse (deceptive practices and confusing practices) by sellers.
  - Requires vendors to make their warranties available pre-sale.
  - Defines some common terms used in warranties.
  - Forbids disclaimer of implied warranty if any written warranty is offered.
  - ***Scope of the statute is consumer goods.***
- Song-Beverly Act (CA) and other State Laws
  - Extend Mag-Moss, for example by saying that disclaimer must actually be provided to ***buyer of consumer goods*** pre-sale. Some of these statutes ban disclaimers of implied warranties in the sale of new merchandise.

Deceptive Practices

- Uniform Deceptive Trade Practices Act:
- A person engages in deceptive trade practices when s/he represents that **goods or services** have sponsorship, approval, characteristics, ingredients, uses, benefits, or quantities that they do not have.
- California Civil Code 1770:
- The following unfair methods of competition and unfair or deceptive acts or practices undertaken by any person in a transaction intended to result or which results in the **sale or lease** of **goods or services** to **any consumer** are unlawful:
  - (a) Passing off goods or services as those of another.
  - (b) Misrepresenting the source, sponsorship, approval, or certification of goods or services.
  - (c) Misrepresenting the affiliation, connection, or association with, or certification by, another.
  - (d) Using deceptive representations or designations of geographic origin in connection with goods or services.
  - (e) Representing that goods or services have sponsorship, approval, characteristics, ingredients, uses, benefits, or quantities which they do not have or that a person has a sponsorship, approval, status, affiliation, or connection which he or she does not have.

Deceptive Practices

California Civil Code 1770:

(f) Representing that goods are original or new if they have deterioriated unreasonably or are altered, reconditioned, reclaimed, used or secondhand.

(g) Representing that goods or services are of a particular standard, quality, or grade, or that goods are of a particular style or model, if they are of another.

(h) Disparaging the goods, services, or business of another by false or misleading representation of fact.

FTC Actions

Go to www.ftc.gov for copies of complaints and settlement agreements involving such companies as Apple, Dell, Gateway 2000, Iomega, and others. Issues include:

- Failure to provide tech support for life after promising to do so
- Failure to disclose key terms of the contract
- Deceptive practices of various kinds

Interesting Questions

Should there be a standard of unfairness for contract terms?

Standard of Unfairness?

- Article 2 adopts a doctrine of unconscionability. Unconscionable terms are typically unfairly surprising. In some states, the terms need only be very harsh, under circumstances where the product was needed and the contract was adhesive (take-it-or-leave-it, no choice, no competition, these are the only terms you can get). UCITA also adopts an unconscionability doctrine, but it is narrower.
- In practice, Article 2's doctrine has been pretty narrow in the courts.
- For some more info, see Carol B. Swanson (2001) Unconscionable Quandary: UCC Article 2 And The Unconscionability Doctrine. See swanson.rtf.
- Article 2 also is subject to a power inherent in the courts, to reject terms that violate public policy. This is not discussed inside Article 2. UCITA makes this explicit, but then (in the eyes of some attorneys) narrows the circumstances under which the judge can rule that a term is sufficiently in violation of public policy to be rejected.
- What (if anything) SHOULD cause us to reject a contract or contract term as unreasonable for the State to enforce?

Interesting Questions

Should embedded and non-embedded software be treated differently in the law?

Embedded Software

- Imagine applying all of the UCITA rules to cars, refridgerators, home heating systems, and anything else that because controlled by computers.
- Would this let manufacturers kill (or take control of) the used car market?
- Would this let manufacturers tell you how you are required to drive their car, who can drive it, where they can drive it, pay an extra fee every time you go to Miami? (Why Miami? Why not?)
- Phil Koopman and I demonstrated repeatedly that the UCITA rules apply to embedded software directly or that manufacturers of goods that have embedded software could make easy engineering changes to get their software under UCITA. Given that, UCITA allows the maker of a mixed-product (some software, some steel) to bring both within the scope of UCITA. Computers and peripherals are singled out for extra easiness this way. See www.badsoftware.com for several papers.

Interesting Questions

To what extent should we rely on electronic signatures?

Interesting Questions

How can software engineers be involved in the legislative and judicial processes?

Engineers in the Process?

- There is room for us in the legislative process.
- We have to be assertive, coherent, and stick to what we know. Engineers who lecture lawyers about law get their noses cut off.
- Engineers who lecture lawyers about engineering get listened to carefully. Specific detail is welcomed by many legislative drafters.
- It's very useful to attend a NCCUSL meeting or another legislative drafting meeting quietly, in preparation for speaking out at subsequent meetings. Many engineers speak at their first meeting and wish they hadn't.
- The legislative drafting process, federally and in the states, has largely been lawyers-only because no one else showed up to the meetings. Over the next 10 years, they'll determine the law and associated culture of our profession. It's worth getting involved.

Useful Legal Research Sites

I give these URLs to my students

Useful Legal Research Sites

- US and State Constitution and Statutes

  http://lcweb.loc.gov/global/judiciary.html

  Links to the United States Constitution and the Federal Statutes and to link pages that get you to the State statutes.

- US Congress

  http://thomas.loc.gov/home/lawsmade.toc.html

  Explains how Federal (U.S.) laws are made.

- US Government Library of Congress

  http://www.loc.gov/

  Link to Copyright Office, many other legislative materials.

Useful Legal Research Sites

- US Government Consumer Law Page

  http://www.consumer.gov

  Multi-agency consumer law page.

- United States Federal Trade Commission

  http://www.ftc.gov

  You might be astonished by who (software / computer companies) has been prosecuted and what they have been prosecuted for.

- American Bar Association (Business Law Section) page on electronic shopping and consumer law.

  http://safeshopping.org/

  This site covers the basics well and carefully.

Useful Legal Research Sites

- John Marshall Law School (Center for Information Technology & Privacy Law)

  http://www.jmls.edu/cyber/index/index.html

  John Marshall publishes the leading journal of computer law. Their index points to a lot of useful research papers.

- UC Berkeley Center for Law & Technology

  http://www.law.berkeley.edu/institutes/bclt/

  Extraordinary scholarship on intellectual property issues and software.

- FindLaw

  http://www.findlaw.com/

  Widely used index to legal materials.

- Guide to Internet Legal Research

  http://www.ali-aba.org/aliaba/search.htm

Useful Legal Research Sites

- National Conference of Commissioners on Uniform State Laws

  http://www.nccusl.org

  This is the organization that authored UCITA and co-authored the Uniform Commercial Code.

- Uniform Commercial Code

  http://www.law.cornell.edu/ucc/ucc.table.html

Useful Legal Research Sites

- National Consumer Law Center

  http://www.consumerlaw.org/

  Leading consumer advocacy group looks at computer law and electronic commerce.

- Consumer Action Handbook

  http://www.pueblo.gsa.gov/crh/respref.htm

  This is an incredibly useful guide and set of contact info for handling / making consumer abuse complaints.

Useful Legal Research Sites

- Software & Information Industry Association

  http://www.siia.net/

  The Software Publishers Association and the Information Industry Alliance merged to form SIIA. This is THE advocacy group for software publishers.

- Business Software Alliance

  http://www.bsa.org/

  Advocates for the really big software companies.

Useful Legal Research Sites

- UCITA advocacy site

  http://www.ucitaonline.com/

  This is the primary advocacy page for UCITA. Selected history and source documents are available at the same author's http://www.2bguide.com/

- 4CITE coalition opposing UCITA

  http://www.4cite.org/

- Bad Software (anti-UCITA) website

  http://www.badsoftware.com

  This is my (Kaner's) page.

- James Huggins UCITA page

  http://www.jamesshuggins.com/h/tek1/ucita.htm

Useful Professional Ethics Sites

- Association for Computing Machinery
  - http://www.acm.org/
  - http://www.acm.org/serving/se_policy/
    - Position on licensing software engineers and on the SWEBOK (Project to draft a Software Engineering Body of Knowledge)
  - http://www.acm.org/serving/
    - Public policy page, links to software engineering code of ethics, copyright policy, free speech.
  - http://www.acm.org/usacm/copyright/
    - Links to US-ACM's position on UCITA

Useful Professional Ethics Sites

- Institute for Electrical & Electronic Engineers
  - http://www.ieee.org
    - Main IEEE page, links to standards, etc.
  - http://www.ieeeusa.org/
  - http://www.ieeeusa.org/forum/index.html
    - IEEE USA public policy committee
  - http://www.computer.org/education/secert.htm
    - Software engineering certification project
  - http://www.swebok.org/
    - Software Engineering “Body of Knowledge”, was a joint IEEE-CS and ACM project, I believe ACM has withdrawn support.

Useful Professional Ethics Sites

- IS World Professional Ethics page
  - http://www.cityu.edu.hk/is/ethics/ethics.htm
  - This is a key page, with lots of links to professional society sites and private sites,copies of codes of ethics from various professions, subscription info for mailing lists, etc.
- Donald Gotterbarn's ethics page / articles
  - http://www-cs.etsu.edu/gotterbarn/articles.htm
- Computer Professionals for Social Responsibility
  - http://www.cpsr.org/
- Web Clearinghouse for Engineering and Computing Ethics
  - http://www4.ncsu.edu/unity/users/j/jherkert/ethicind.html

Useful Professional Ethics Sites

- The RISKS digests
  - http://catless.ncl.ac.uk/Risks
    - Lots of great examples and gossip.
- WWW Resources for Professor Kevin Bowyer's Ethics & Computing Course
  - http://marathon.csee.usf.edu/~kwb/ethics-and-computing.html
    - Good stuff, great links.

References

References on the Disk

- Alces, Peter & David Frisch (1998), Commercial Codification as Negotiation. See alces.rtf.
- American Bar Association Working Group Report On The Uniform Computer Information Transactions Act, See aba_ucita.pdf
- Arizona Retail Systems v. The Software Link (1993) United States District Court For The District Of Arizona. *See arizona_retail.rtf*
- Braucher, Jean (2000), Delayed Disclosure in Consumer E-commerce as an Unfair And Deceptive Practice. See braucher.rtf.
- Brennan, Lorin (2000), Symposium on Approaching E-commerce Through Uniform Legislation: Understanding the Uniform Computer Information Transactions Act and the Uniform Electronic Transactions Act: Why Article 2 Cannot Apply To Software Transactions. See brennan.rtf.
- Carroll, Darren J. (1992) When More Is Less: Controlling The Market For Computer Software Enhancements. See carroll.rtf.
- Determann, Lothar & Aaron Xavier Fellmeth (2001), Don't Judge a Sale by Its License: Software Transfers Under the First Sale Doctrine in the United States and the European Community, See determann.rtf.
- Dodd, Jeff C.(1999), Time And Assent In The Formation Of Information Contracts: The Mischief Of Applying Article 2 To Information Contracts, See dodd.rtf.
- Dreyfuss, Rochelle Cooper (1999), Do You Want to Know a Trade Secret? How Article 2B Will Make Licensing Trade Secrets Easier (But Innovation More Difficult). *See dreyfuss.rtf.*
- Foster, Ed, John Fontana, John, and Steve Murchie, Articles (and MS Response) on Microsoft's refusal to allow publication of benchmarks. *See microsoft_forbids_benchmarks.rtf*
- Gomulkiewicz, Robert W. (1998), The License Is The Product: Comments on the Promise of Article 2B for Software and Information Licensing. See gomulkiewicz.rtf
- Georgenson, Karen (1996), Reverse Engineering of Copyrighted Software: Fair Use Or Misuse? See georgenson.rtf.
- Hillman, Robert. (1999) The "New Conservatism" in Contract Law and the Process of Legal Change. See hillman.rtf.
- Hoye, James (2001), Click – Do We Have a Deal? See hoye.rtf.
- I.LAN SYSTEMS V. NETSCOUT SERVICE LEVEL Corp. (2002), United States District Court For The District Of Massachusetts. *See ilan.rtf*
- Ionics Inc. v. Elmwood Sensors (1997) United States Court Of Appeals For The First Circuit. *See ionics.rtf.*
- Kaner, Cem (1995) Liability for Defective Documentation, *defective_doc.rtf.*
- Kaner, Cem (1996) Liability for Defective Content, *kaner_content.rtf*
- Kaner, Cem (1996), Quality / Cost Analysis: Benefits and Risks. See kaner_qualcost.rtf
- Kaner, Cem (1996, updated 2000) Computer Malpractice. See kaner_malprac.rtf.
- Kaner, Cem (1998), The Problem of Reverse Engineering, See kaner_revengr.rtf.
- Kaner, Cem (2000), Software Engineering & UCITA. See kaner_sweucita.rtf
- Kaner, Cem (2001, September) Software Engineering as a Profession After the Withdrawal: One Year Later (Position Statement). See kaner_fase.rtf

References on the Disk

- Karas, Stan (2001) INTELLECTUAL PROPERTY A. COPYRIGHT 2. DEFENSES A) FAIR USE: Sony Computer Entertainment, Inc. v. Connectix Corp. *see karas.rtf*
- Klocek v. Gateway 2000 (2000) United States District Court For The District Of Kansas. *See klocek.rtf.*
- Lemley, Mark (1995), Intellectual Property and Shrinkwrap Licenses. *See lemley.rtf*
- Lewis, Terril (2000), Reverse Engineering Of Software: An Assessment Of The Legality Of Intermediate Copying. *See lewis.rtf.*
- M.A. Mortenson Company, Inc., Petitioner, V. Timberline Software Corporation, Supreme Court of the State of Washington, *see mortenson.rtf*
- Meiklejohn, Alexander (1994), Castles in The Air: Blanket Assent and The Revision of Article 2. See Meiklejohn.rtf.
- National Association of Attorneys General (33 Attorneys-General) (2001), Re: The Uniform Computer Information Transactions Act. See naag.pdf.
- National Conference of Commissioners on Uniform State Laws, (2001) Uniform Computer Information Transactions Act, see ucita2001.rtf
- NCCUSL Standby (Drafting) Committee for UCITA, proposed amendments to UCITA, See ucitaamend.rtf.
- Nimmer, David , Elliot Brown, & Gary N. Frischling (1999) The Metamorphosis of Contract into Expand. *See david_nimmer.rtf*
- Nimmer, Raymond (2000), Symposium On Approaching E-commerce Through Uniform Legislation: Understanding The Uniform Computer Information Transactions Act And The Uniform Electronic Transactions Act: Through the Looking Glass: What Courts and UCITA Say About the Scope of Contract Law in the Information Age. See nimmer.rtf.
- People of the State of New York v. Network Associates (2002), Memorandum of Law in Support of Petition of Attorney-General, *see nyntwrkass020702mol.pdf* and Petition, *see nyntwrkass020702pet.pdf*
- Selman, Jeffrey C. & Christopher S. Chen (1997), Steering the Titanic Clear of the Iceberg: Saving the Sale of Software from the Perils of Warranties. See selman.rtf
- Carol B. Swanson (2001) Unconscionable Quandary: UCC Article 2 And The Unconscionability Doctrine. See swanson.rtf.
- Towle, Holly K. (1999) The Politics of Licensing Law. See towle.rtf
- Uniform Commercial Code (as enacted by Connecticut), See tp42a-p1.htm.
- Universal Studios v. Corley (1/29/02), United States Court of Appeals for the Second Circuit, *see Universal2002.rtf*
- Universal Studios v. Remerdes (8/17/2000) United States District Court for the Southern District of New York, *see Universal0800.rtf*