The UCITA Battle:

Should it be legal for software to be defective?

Prof. Philip Koopman

Carnegie Mellon University

koopman@cmu.edu

March 2002

Electrical & Computer
ENGINEERING

Carnegie Mellon

Overview

- **Q: What's the cheapest way to deal with SW defects?**
  - A: Get a law passed saying that you don't have to fix them.

- **Background: UCC, UCITA & Mag./Moss**
  - Why having something special for software might make sense

- **Problems with UCITA**
  - Why embedded systems spell the downfall of UCITA as it stands
    - (watch closely here for Genuine Technical Content!)
  - The larger issue: does software really have to work?
    - Desktop software?
    - Safety critical embedded software?
    - Systems that have both kinds of software?

UCC 2 & UCITA

Prologue: UCC 2

- **Uniform Commercial Code, Article 2** (sales of goods)
- **Existing consumer protections via Magnuson/Moss**
  - "Plain English," clear & understandable warranty text
  - Warranty can't disclaim/limit rights provided by state law
  - Warranty text must be available before the sale
  - Vendors must honor warranties
    - You have the right to a refund if you return defective software promptly

- **Implications for software**
  - "Shrink wrap" licenses historically have been unenforceable
    - Lawyers like to put in unenforceable terms as a negotiating tactic
  - Software in "goods" is still "goods"
    - i.e., it is supposed to provide all *advertised features*

Carnegie Mellon

UCC Is A "Uniform State Law"

◆ **Central organization drafts Model Acts**

- National Conference of Commissioners on Uniform State Laws
- Volunteer lawyers from each state participate
  - No techies officially involved; *none*; no kidding

◆ **NCCUSL just recommends Acts**

- Each state legislature must pass a bill to make it a state law
- Typically Acts are passed without modifications, but not always

◆ **In practice, others become involved, especially**

- American Law Institute (ALI) – drafting partner in UCC
- American Bar Assn. (ABA) – acts as independent endorser

Carnegie
Mellon

The Problem They're Trying To Solve

◆ **Software represents significant liability risk**

- In general, the software industry isn't able to get software "right"
  - Market pressures & business models force hasty releases
  - Software is very labor intensive and expensive
- Consumers historically vote with $$$ for features, not reliability
  - Then they press class-action suits when software is flaky(!)

◆ **Intuitively, perhaps desktop software is different**

- Key proponent idea:
  - Desktop software gets sold "as is"
  - "Embedded software" is still "goods", and is not as-is
- Goal of new Act: *default terms* for software contracts

First Attempt: UCC Article 2B

◆ **Addition to Article 2 for software**

- 1995: NCCUSL & ALI jointly drafted

- 1998: ALI withdrew support due to bias in favor of SW industry
  - This was an unprecedented move

◆ **So a new drafting committee was created**

- 1999: NCCUSL-only committee

Carnegie
Mellon

Round 2: UCITA

◆ **Uniform Computer Information Transaction Act**

- Supported by AOL/Time-Warner, Microsoft, Lexis (database publisher), computer manufacturers, and even Chrysler
- Book & entertainment/movie industries got "carve-outs"

◆ **Lots of people are against current Act, including:**

- 34 Attorneys General   (up from 26 a year ago)
- American Library Association
- ACM & IEEE
- Free Software Foundation
- Every consumer advocacy group that has considered it
- SEI
- Life insurance companies; some manufacturers

**Carnegie
Mellon**

What Terms Are Problematic?

◆ **Default rules seem anti-consumer:**

- "Click-wrapped" agreements that are not visible before the sale
  - You don't get to see the agreement until you are at home
    - » Know any stores that will take back opened software without a hassle?
  - Significant restrictions can be revealed after the sale (FTC opposes this)

- License terms can change after the purchase
  - They just have to send an e-mail or post on a web site
  - We're talking *any* change, however major

- Can't transfer license of software
  - Even for corporate merger, or change of position within company(!)
  - Libraries, resale of used software, *etc.* could be restricted
  - Fundamentally undermines copyright fair use doctrine

Carnegie Mellon

More UCITA Issues

◆ **Legitimizes current dubious practices:**

- Prohibitions on reviews and benchmarks
  - If it doesn't work, you can't publish an article saying it is broken

- Usage tracking to detect license violations
  - Vendor can send info from your PC with no predetermined limitation

- Implied warranties can be disclaimed
  - No duty to warn of material known defects
  - With usual terms, the software doesn't have to work at all!

- Arbitration at any worldwide location can be specified
  - That one might not hold up in court, but that battle still must be fought

Carnegie Mellon

Even More UCITA Issues

◆ **Significant quality/security concerns**

- Can prohibit reverse engineering
  - Even for security/interoperability analysis
- No duty to check for viruses on distribution media
- "Self help" back-doors for electronic repossession/shutdown
  - *Without* warning; without court order
  - For pretty much *any* reason they contend is a consumer license violation
    - » Contract can remain enforceable if the vendor breaches it, however
  - (*You* are *not* the one being "helped")

◆ **"Opt-in" provisions if software is a "material" factor**

- Means entire sale is under UCITA, including non-SW "goods"
  - Might be applicable to cars, pacemakers, etc.
  - This is incentive to add software to products!

Carnegie Mellon

Academic Issues

◆ **Academics will have problems**

- Be sure to put a click-wrap agreement in any source code downloads, then hope no problems develop

◆ **Tricky risk inherent in publishing code**

- Code published in a book is exempt from UCITA
  - Book publishers have obtained a UCITA carve-out, so they are safe
- But, as soon as someone compiles the code, UCITA covers it
  - That's right, even if they type it in from a book
  - You have no control over this situation, but you are liable for the full default terms of UCITA if you don't have a "contract" waiver in place
    » But there is no known effective way to put a contract in place

UCITA Is Already Law

◆ **Virginia & Maryland have passed slightly modified versions**

◆ **That means UCITA matters to YOU**
  - UCITA permits vendors to bring contract under those laws…
    … even if you don't live in those states
  - So you are now subject to UCITA if vendors really want to do it

◆ **Except in "bomb shelter" states**
  - Iowa, West Virginia & North Carolina have "UCITA bomb shelter" laws to protect their citizens
    – (No, you can't invoke them as a consumer in PA)

Carnegie
Mellon

Embedded Software: A Key Battleground

Software Warranty Workshop

◆ **Sponsored by Federal Trade Commission**

- October 2000, Washington DC

◆ **Both sides represented**

- Lawyers for UCITA
- Lawyers against UCITA
- State & federal government representatives & staffers
- One engineer that I know of (me); invited by FTC

◆ **FTC is currently dormant**

- Went quiet right after Bush was elected
  – May be getting back into the game … it will take time

Carnegie
Mellon

Licenses For Embedded Systems?

- ◆ **Current protection based on patents**
  - In embedded systems, *functionality* is what matters, not "software" vs. "hardware" (in fact, they can be equivalent)
  - Functionality can be patented, and has been for decades
  - Software can be patented too; ROM images/masks copyrighted

- ◆ **Opinion: Encouraging embedded software licensing is potentially dangerous**
  - Currently, embedded software is not considered "special"
    - This moderates the rate of introducing new features
    - This is one of the few forces acting to moderate the software safety problem (we're still struggling with how to measure "software safety")
  - Do you really want embedded software to be as non-robust as current desktop software?

Carnegie Mellon

Would *You* Drive A Car In Which:

"THE SOFTWARE is provided 'AS IS' and with all faults. THE ENTIRE RISK AS TO SATISFACTORY QUALITY, PERFORMANCE, ACCURACY, AND EFFORT (INCLUDING LACK OF NEGLIGENCE) IS WITH YOU."

(You will.)

Embedded OS License Survey

◆ **Company A:** (License wording available on the Web)
- Any use constitutes agreement
- No Warranties; "As is" and with all faults and any negligence
- Any user of product of which it is a component must agree
- Reverse engineering prohibited

◆ **Company B:** (License wording available on the Web)
- Same as above, PLUS
- Leasing or sale of software prohibited; can't leave country
- "Bugs are likely"
- But, warrants it will work per documentation for 6 months

◆ **Company C:** flatly refused to provide EULA

Carnegie
Mellon

My FTC Talk Conclusions:

◆ **Fundamental problems:**

- "Embedded" computers and "computers" are converging
  - Any potentially useful definition can be discredited or circumvented
- The concept of saying "software is different" is dangerous
  - Converting complexity into software instead of hardware is easy
- Consumers will be hurt by licensing embedded software
  - This is already happening; it just hasn't reached high market penetration yet

◆ **Embedded software licensing is going to be a mess**

- UCITA/UCC wording requires significant fixes; may be unfixable
- Even if "embedded" can be excluded from UCITA, there will be compelling incentive to make everything look like a non-embedded "computer"

Carnegie Mellon

The Battle Of The Wording

◆ **Early 2001:**
Several iterations for attempted UCITA workarounds

   • Nobody at the FTC workshop dared admit embedded SW would be in scope

   • Attempted to make split with revision to UCC 2

   • Attempted to propose alternate UCITA wordings

◆ **Ultimately, it is impossible to separate Embedded and desktop software**

   • *IF* there is *incentive to change*, any differentiator will evaporate

UCITA Wording – 1

(1) If a transaction includes computer information and goods, this [Act] applies to the part of the transaction involving computer information, informational rights in it, and creation or modification of it. However, if a copy of a computer program is contained in and sold or leased as part of goods, this [Act] applies to the copy and the computer program only if:

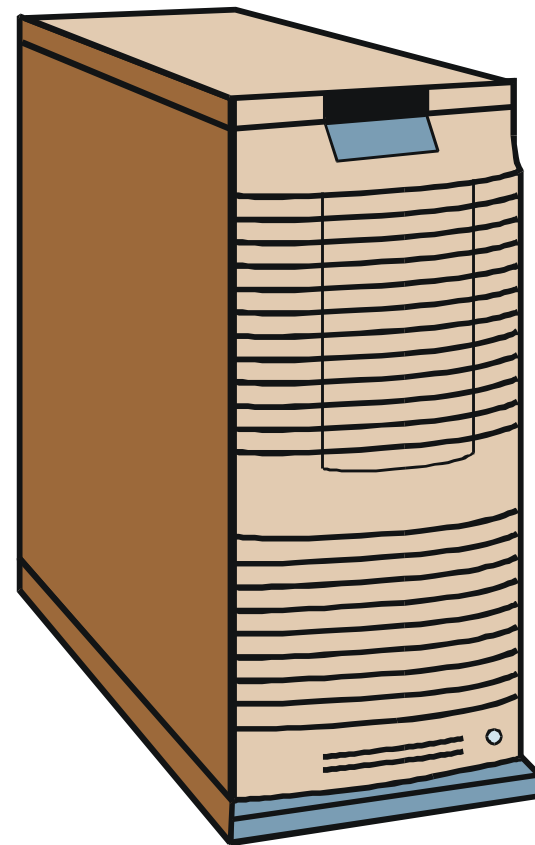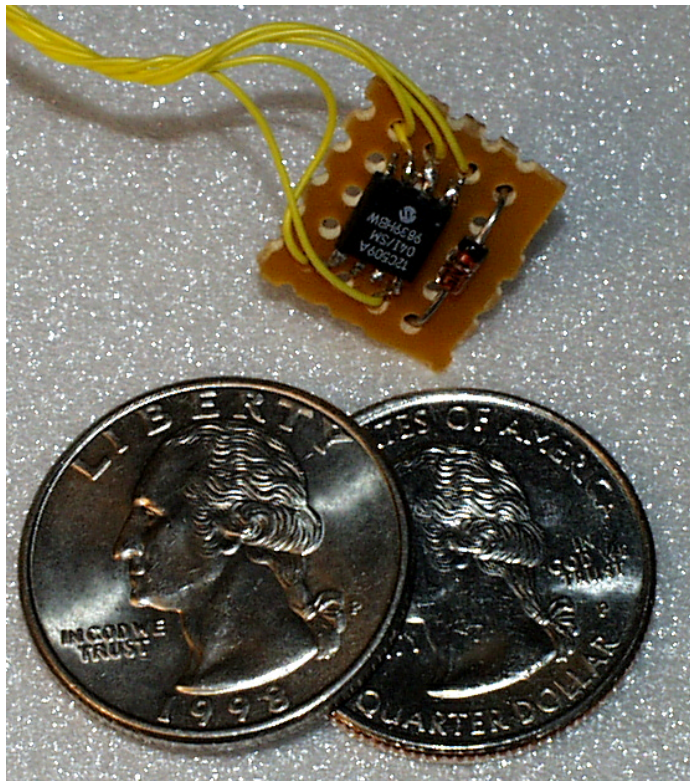   (A) the goods are a computer or computer peripheral; or

        (B) giving the buyer or lessee of the goods access to or use of the program is ordinarily a material purpose of transactions in goods of the type sold or leased.

These Are Both Web Servers

◆ **Which one is a "computer"?**

  • They're both "computers", even if one is in a thermostat.

http://www-ccs.cs.umass.edu/
~shri/iPic.html/

UCITA Wording – 2

◆ **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.**

- *This definition applies to almost any possible computer, embedded or otherwise* (*anything with a program counter*)

- *The discussions given in the official comments are based on incorrect assumptions, are contradictory, and do not reflect the actual UCITA wording*

- *"Computer peripheral" is undefined in UCITA*

NUCCSL Attempted To Fix Things

◆ **Official comments are the non-binding rationale**

- (By analogy, the Act is the specification)
- Attempted to fix by roughly arguing "you know what I mean"

◆ **Argued that a sewing machine was a classical embedded system**

- Asserted that any computer in a sewing machine would be clearly hidden
- Functionality would be fixed at time of sale
- *Etc.*

Husqvarna Viking Designer

◆ **Sewing machine or computer?**

- Even has a bug patch download Web page!

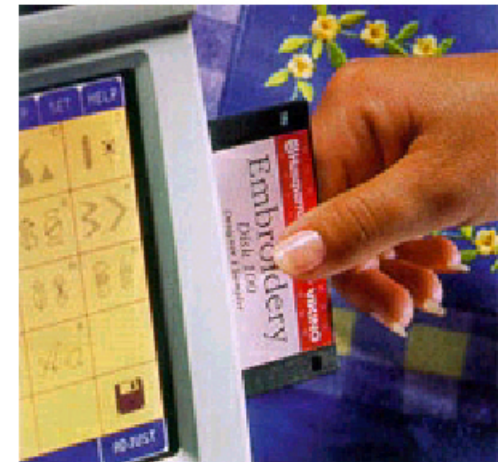Husqvarna Viking Designer I update Rev 1.33

▶ Download

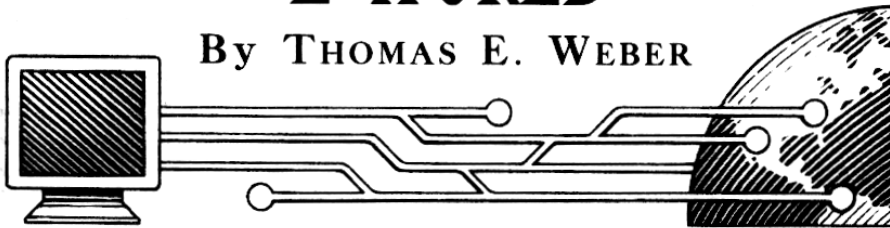Color Touch Screen

Sewing Advisor

Built-in Floppy Disk Drive

Carnegie Mellon

E-WORLD

By THOMAS E. WEBER

Software-Warranty Law Spurs Fears We'll Soon See Bugs Everywhere

LOOK AROUND at the conveniences of every-day life. Check out your car, your cellphone, maybe your wristwatch. Can you tell what's a computer and what's not?

. . . .

. . . .

"I'm extremely concerned," says Philip Koopman, a professor of electrical and computer engineering at Pittsburgh's Carnegie Mellon University. "There's a tremendous risk of adding complexity and reducing reliability in all kinds of products."

Prof. Koopman is an expert on "embedded systems," which means computers that are built into things like elevators and air conditioners. He fears that Ucita, by setting up rules for software warranties that would be different and potentially less stringent than the rules covering other products, would create an economic incentive for manufacturers to use software wherever possible to reduce their liability. If that happens, Prof. Koopman says, we could start to see more and more every-day devices becoming as balky as our desktop PCs.

UCITA questions could lead to purposefully poor design of ordinary goods

- Interview with P. Koopman from Carnegie Mellon:
  - Optimization for legal liability instead of sound engineering judgment
  - Migration of dangerous functions to software so liability can be disclaimed

◆ **Companion thought piece published**

- Spins a nightmare scenario based on "what if tires had software?"
- Concludes it's a good thing that tires don't have software now that UCITA is around …. but ….

Carnegie Mellon

Some Tires Already Have Software

The key components of the remote sensing module for the TPMS are the MPXY8020A6 temperature and pressure sensor, a single-chip circuit with a a wakeup feature; and the MC68HC908RF2, consisting of the HC08 microcontroller and an RF transmitter-two chips in one package.

Carnegie
Mellon

Cycle Repeated With Various Problems

◆ **Summary 2001 UCITA comments example: washing machine**

Surf Among Suds With Web-Enabled Washing Machine

LG Electronics unveils its second Internet-aware appliance, which downloads clothing care programs.

Martyn Williams, IDG News Service
Tuesday, October 17, 2000

Is This Cell Phone A Computer?

- **Availability:** Europe & Africa, Asia
- **Weight:** 244 g
- **Dimensions:** 158 x 56 x 27 mm
- **Talk time:** 4 - 10 h
- **Standby time:** up to 230 h

Carnegie
Mellon

Is It Now?

NOKIA
CONNECTING PEOPLE

Surprisingly Effective

NOKIA 9210 Communicator

The Nokia 9210 Communicator at a Glance

- **Availability:** Europe & Africa, Asia
- **Weight:** 244 g
- **Dimensions:** 158 x 56 x 27 mm
- **Talk time:** 4 - 10 h
- **Standby time:** up to 230 h
- **Key features:** Full-color screen with 4096 colors, Mobile email, Word, Spreadsheet, Presentation viewer, High speed WWW and WAP, Mobile multimedia.
- **Operating frequency:** EGSM 900/1800 networks in Europe & Africa, Asia

Embedded Internet Systems?

◆ **What if you use "desktop" software on a desktop**
- And it controls a safety critical embedded system?
  - Does the desktop software have to work?
  - If it doesn't work and someone dies, can the warranty disclaim liability?

◆ **Easy horror stories to hypothesize**
- "Secret" feature for monitoring/maintenance leaked on WWW
  - Thieves can enter their own access code and steal computer equipment
  - Perhaps they can take down hospital primary + backup electricity
- House manager computer closes garage door if you forget
  - Interacts with software to pre-start car, causing carbon monoxide poisoning
- Automobiles
  - What happens if WWW connects to automotive control systems?
- Internet pacemaker

Why Can't This Be Pinned Down?

◆ **They tried saying: computer "incorporated" into goods**

- Dynamic vs. fixed functionality   (just in time compilers??)

- Data vs. program (interpreted languages??)

- Can be programmed (like a setback thermostat??)

- Runs arbitrary programs (what if Tivo shows its Linux prompt??)

- Software sold separately
  - Your car and DVD player get bug patches; why not sell SW separately?

- HP 9100C Digital sender – computer or scanner?

- *(Are gate definitions for an FPGA software or hardware?)*

Carnegie Mellon

Dr. Turing To The Rescue

◆ **Koopman & Kaner argue to Turing Equivalence**

- 3-part article in UCC Bulletin Feb-Apr 2001
  - Widely read & influential publication in that arena
  - (Well, OK, it's really Thesis M, but people have heard of Turing…)

- Thesis M: *Whatever can be calculated by a machine is Turing-machine-computable.*

◆ **The implication is:**

- If there is a reason to make an embedded computer look/feel/smell/taste like a desktop computer, then someone will probably do that, and in theory they can *always* do that
- But now we are arguing to bedrock CS principles, not just ad hoc examples

Mixed Results From Turing Argument

◆ **UCC Article 2 committee appeared to be swayed**

- On-the-fence committee members stopped supporting UCITA
- Software exclusion for UCC 2 seems to have been defeated
    - That won't stop lobbying money from being spent of course
- That left only NUCCSL and ABA to convince

◆ **Fall 2001:
UCITA committee gives up on rewording and pushes on**

- Definition of "computer" still broken
- Major problems remained
- But the train was still chugging along

Carnegie
Mellon

The Big ABA Showdown

◆ **November 2001 – first public hearings with ABA**

- ABA members present to decide whether to endorse UCITA
- Large array of forces present (200+ people) in Washington DC
- A handful of engineers (one each: IEEE, ACM, CMU)
- New proposal: exclude "safety critical" apps – NUCCSL refused

◆ **Public comments and debate on proposed changes**

- Partly for NCCUSL to consider changes
- Just as much for ABA to hear both sides of problems

Carnegie Mellon

ABA Says No Thanks

◆ **ABA conclusion: they can't support UCITA**

- Very harsh criticism; embedded issues figured prominently
  - Basically said it needed to be re-written to be understandable(!)

◆ **Latest UCITA revision since then**

- Appears to drop "self-help"
  - This might please the insurance industry
- More or less business as usual EXCEPT …

  … no 3rd party organizations are left on their side
- Future is uncertain
  - But there is too much lobbying money for this to just die

Techies Can't Afford To Ignore Things!

◆ **Key champion against UCITA: Cem Kaner**

- Ph.D. Computer Science
  - Specializes in software testing
  - Now a CS prof. at FIT
- Lawyer (J.D.) too
  - Book on law & software quality
- Has made numerous and significant personal sacrifices to fight UCITA, and it has paid off

◆ **More champions will be needed next year/decade (this means you)**

- Do most lawyers understand technology?
- Can they regulate what they don't understand?

Carnegie Mellon

Pointers

◆ **Best single paper:** Kaner, Software Engineering & UCITA
  - http://www.badsoftware.com/engr2000.htm

◆ **General UCITA info (pro-UCITA editorial content)**
  - http://www.ucitaonline.com/

◆ **Top 12 Consumer Problems with UCITA**
  - http://www.consumerlaw.org/ucita/twelve_problems.html

◆ **Dig deeper into some of the issues:**
  - http://www.badsoftware.com/uccindex.htm

◆ **Anti-UCITA consumer group:**
  - http://www.4cite.org/

Carnegie Mellon