computer reliability

Burak Galip ASLAN, PhD

introduction

trivial computer errors -> game, homework...

introduction

trivial computer errors -> game, homework...

inconvenient computer errors -> incorrect billing, software bugs, fatalities...

introduction

trivial computer errors -> game, homework...

inconvenient computer errors -> incorrect billing, software bugs, fatalities...

systems typically have many components, of which the computer is just one

introduction

trivial computer errors -> game, homework...

inconvenient computer errors -> incorrect billing, software bugs, fatalities...

systems typically have many components, of which the computer is just one

a well-engineered system can tolerate the malfunction of any single component **without failing**

introduction

trivial computer errors -> game, homework...

inconvenient computer errors -> incorrect billing, software bugs, fatalities...

systems typically have many components, of which the computer is just one

a well-engineered system can tolerate the malfunction of any single component **without failing**

computer -> usually the weakest link in system ... computer failures led to **software engineering** science

data-entry or data-retrieval errors

failure because of wrong data entered or retrived data interpreted wrongly

data-entry or data-retrieval errors

failure because of wrong data entered or retrived data interpreted wrongly

disfranchised voters: 2000, general election Florida, thousands of voters are disqualified due to a database error, this error might have affected the outcome of the election

data-entry or data-retrieval errors

false arrests: 40 million records, stolen automobiles, missing persons, wanted persons, suspected terrorists and more

data-entry or data-retrieval errors

false arrests: 40 million records, stolen automobiles, missing persons, wanted persons, suspected terrorists and more

- Sheila Jackson Stossier, airline flight attendant, arrested in New Orleans airport, Shirley Jackson wanted in Texas, 1 night in jail + 5 days detention

data-entry or data-retrieval errors

false arrests: 40 million records, stolen automobiles, missing persons, wanted persons, suspected terrorists and more

- Michigan resident Terry Dean Rogan personal info used for obtaining a California driver's license, person is arrested for 2 homicides and 2 robberies, crimes recorded under false identity, a period of 14 months, real person arrested 5 times by LAPD of which three times gun point even though Michigan police corrected records after first arrest, Rogan sued LAPD and won $55000

data-entry or data-retrieval errors

analysis: accuracy of NCIC (National Crime Information Center) records

data-entry or data-retrieval errors

analysis: accuracy of NCIC (National Crime Information Center) records

step away from Privacy Act (1974), 2003 Justice Dept. announced FBI no longer ensures accuracy of info about criminals and victims before entering it in NCIC db

data-entry or data-retrieval errors

analysis: accuracy of NCIC (National Crime Information Center) records

step away from Privacy Act (1974), 2003 Justice Dept. announced FBI no longer ensures accuracy of info about criminals and victims before entering it in NCIC db

Dept. of Justice argues that it is impractical to be responsible for every single data in NCIC db, information sources are very diverse, FBI has no way of verifying, agents use discretion, if verified the db infos will be limited, much less useful tool, less criminals caught

data-entry or data-retrieval errors

privacy advocates strongly counter that the accuracy of NCIC databases is more important than ever now (false arrest issue)

data-entry or data-retrieval errors

privacy advocates strongly counter that the accuracy of NCIC databases is more important than ever now (false arrest issue)

which argument is stronger?

data-entry or data-retrieval errors

privacy advocates strongly counter that the accuracy of NCIC databases is more important than ever now (false arrest issue)

which argument is stronger?

one of the oldest DBs, the database of stolen vehicles, 1 million stolen automobiles / year in US, victims harmed, everyone harmed because of raising insurance costs, state-crossing had been diminished by central db, NCIC increased car recovery by +10%, 50000 additional cars returned each year, few false arrests each year, benefit > harm

software and billing errors

even if data entered is correct, system may still produce wrong result or even may collapse entirely, newspapers are full of stories of bugs and glitches

software and billing errors

- amazon.com, British website, March 13, 2003, sw error, iPaq handheld computers, 7 euro instead of 275 euro, before shutdown bargain hunters flocked to amazon.com some even ordering tens of devices, amazon requested difference cost for delivery

errors leading to system malfunction

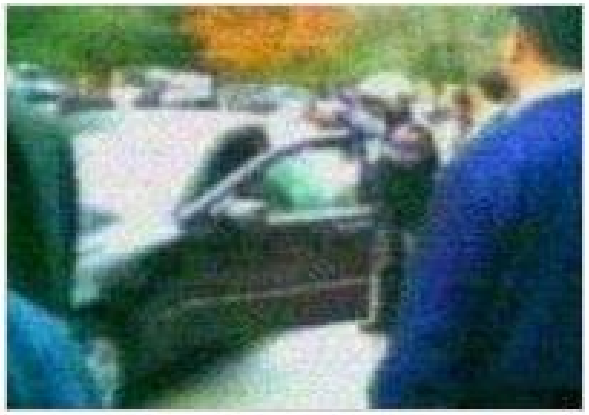
- Linda Brooks, Minneapolis, phone bill of +57,346.20, Qwest billing sw bug, some customers are charged for $600/min for cell phones, 1.4% of Qwest customers -> 14000 customers received incorrect bills, the bug was in the newly installed billing system

errors leading to system malfunction

- University of Pittsburgh study, for most students, computer spelling and grammar checkers increasing errors

errors leading to system malfunction

ERDOĞAN ARACINDA NEDEN MAHSUR KALDI?



17 Ekim 2006 23:06

Başbakan, tam baygınlık geçirdiği sırada makam aracında mahsur kaldı. Peki sorumlu kim?

- Thailand's finance minister trapped inside BMW limousine for 10 mins when on-board computer system crashed, locking all doors and turning off air conditioning, solution: sledgehammer to window

errors leading to system failures

- new laboratory computer system at LA medical center become backlogged the day after it was turned on, for two days emergency doctors stopped ambulance services because they couldn't reach laboratory results... Dr. Amanda Garner: **"We rely so much on our computers and our fast-world technology that we were almost blinded."**

errors leading to system failures

- 1998, sw error, Chicago Board of Trade, suspending trading for 1 hr, 45 minutes a few months later, some investors lost money, same trouble happened London IFFO exchange twice within two weeks of 1999

errors leading to system failures

- Comair (subs Delta Airlines) cancelled all of its flights (1100) christmas 2004, computer system assigning crews to flights stopped running, sw couldn’t handle large number of flight cancellations, 30000 travelers in 118 cities were affected

Comair Cancelled All Flights on Christmas Day, 2004

AP Photo/Al Behrman, File

errors leading to system failures

- 2005, Malaysia Airlines, Perth -> Kuala Lumpur, roller-coaster ride 7 miles above the ocean, Boeing 777 rapid climb, pilot disconnectes autopilot but there's 45 seconds delay, up-down-up-level out, sw error faulty information about plane's speed and acceleration, another error also causing delay in auto-pilot disabling

errors leading to system failures

Amsterdam'da düşen THY uçağının suçlusu Boeing

hurriyet.com.tr / DIŞ HABERLER 6 Mayıs 2010

Hollanda'nın Amsterdam kentinde Şubat 2009'da meydana gelen uçak kazasında büyük oranda üretici firma Boeing'in kusurlu olduğu tespit edildi. Kazada pilotların da kısmen hatalı olduğunu belirten rapora THY tepki gösterdi.

- AÇIKLANAN RAPORA THY'DEN İLK TEPKİ
- DÜŞEN THY UÇAĞININ FOTOĞRAFLARI

İstanbul-Amsterdam seferini yapan THY uçağının iniş sırasında düşmesi sonucunda 3'ü pilot 9 kişi hayatını kaybetmiş, 84 kişi de yaralanmıştı.

Hollanda Havacılılık Emniyet Kurulu kazayla ilgili yaptığı inceleme sonucunda nihai

- another case, 2 altimeters (to cross-check), “genious” solution “poor” application, the accurate one is not working, the inaccurate one is working... (need verification of course)

notable software system failures

- embedded system -> computer used as a component of a larger system
- hw controllers are being replaced by microprocessors controlled by sw -> sw controllers faster, perform sophisticated tasks, manipulating more data, cost less, use less energy, do not wear out
- hw controllers high reliability -> sw controllers not quite high
- most embedded systems are real-time systems with sensors (airbags etc.)

Patriot missile

- 1991 Gulf War, US Army invention, defending against Scud missiles launched at Israel and Saudi Arabia

Patriot missile

- 1991 Gulf War, US Army invention, defending against Scud missiles launched at Israel and Saudi Arabia
- end of Gulf War, patriot system declared as 95% effective at destroying Scuds, later analysis -> only 9% of Scuds were actually destroyed by Patriot missiles, most Scuds are poorly designed and fall apart approaching their target :p

Nedenlerden üçüncüsü, vurucu gücü ne olursa olsun, tek bir silaha dayanmanın yarattığı aşırı ve yapay güven duygusudur. Saddam, Sovyetler'den aldığı Scud füzelerine ve bu füzelerin ucuna yerleştirmeyi planladığı kimyasal/biyolojik başlıklara güveniyordu. Ancak, bu füzeler savaş sırasında istenilen başarıyı gösteremedi. Füzeler Amerikan Patriot Hava Savunma sistemi tarafından havada yok edildiler.

Patriot missile

- 1991 Gulf War, US Army invention, defending against Scud missiles launched at Israel and Saudi Arabia
- end of Gulf War, patriot system declared as 95% effective at destroying Scuds, later analysis -> only 9% of Scuds were actually destroyed by Patriot missiles, most Scuds are poorly designed and fall apart approaching their target :p
- Feb 25, 1991, a Scud from Iraq hit a US barrack at Saudi Arabia killing 28 soldiers, Patriot didn’t even fired...

Patriot missile

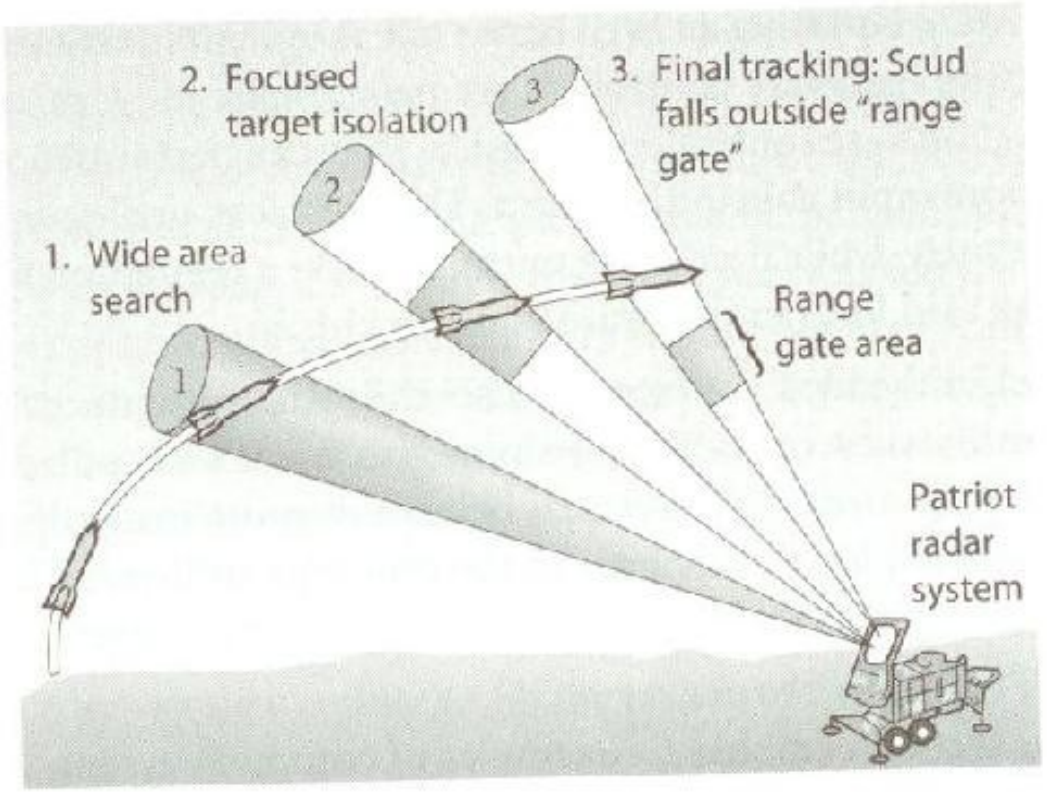


FIGURE 7.1 A software error caused the Patriot missile system to lose track of incoming Scud missiles. (1) The radar system doing a wide area search picks up the Scud missile. (2) The radar system isolates the proposed target. (3) A software error causes the system to produce a faulty range gate. The system loses track of the missile, because it does not fly through this gate. (Reprinted with permission from Marshall, SCIENCE 255:1342 (1992). Illustration: D. Defrancesco. Copyright 1992 AAAS.)

Patriot missile

multi-check system for false alarms, range-tracking multiple checks

range values stored at a floating point variable, round-off errors are added up during system run, the longer system runs, the round-off errors grow, estimated for a few hours runtime, however the system was operating for more than 100 hours, the accumulation of errors lead to 0.3433 second difference between actual and computed time leading to a tracking error of 687 meters...

Patriot missile

Q: What is round off error?

A: A computer can only represent a number approximately. For example, a number like $\frac{1}{3}$ may be represented as 0.333333 on a PC. Then the round off error in this case is $\frac{1}{3} - 0.333333 = 0.0000003\overline{3}$. Then there are other numbers that cannot be represented exactly. For example, π and $\sqrt{2}$ are numbers that need to be approximated in computer calculations.

Q: What problems can be created by round off errors?

A: Twenty-eight Americans were killed on February 25, 1991. An Iraqi Scud hit the Army barracks in Dhahran, Saudi Arabia. The patriot defense system had failed to track and intercept the Scud. What was the cause for this failure?

The Patriot defense system consists of an electronic detection device called the range gate. It calculates the area in the air space where it should look for a Scud. To find out where it should aim next, it calculates the velocity of the Scud and the last time the radar detected the Scud. Time is saved in a register that has 24 bits length. Since the internal clock of the system is measured for every one-tenth of a second, 1/10 is expressed in a 24 bit-register as 0.000110011001100110011001100. However, this is not an exact representation. In fact, it would need infinite numbers of bits to represent 1/10 exactly. So, the error in the representation in decimal format is

Patriot missile

$$\frac{1}{10}-(0\times 2^{-1}+0\times 2^{-2}+0\times 2^{-3}+1\times 2^{-4}+\ldots+1\times 2^{-22}+0\times 2^{-23}+0\times 2^{-24})$$

$$=9.537\times 10^{-8}$$

The battery was on for 100 consecutive hours, hence causing an inaccuracy of

$$=9.537\times 10^{-8}\,\frac{\text{s}}{0.1\text{s}}\times 100\,\text{hr}\times\frac{3600\text{s}}{1\text{hr}}$$

$$=0.3433\text{s}$$

The shift calculated in the range gate due to 0.3433s was calculated as 687m. For the Patriot missile defense system, the target is considered out of range if the shift was going to more than 137m.

Ariane 5

satellite launch vehice, European Space Agency, maiden flight on June 4, 1996, 40 seconds after sw error boosters and main rocket engine swivel to extreme positions, sharply off-course, core and booster lost contact, vehicle self-destructs itself, satellites carried $500 million were not insured

Ariane 5

satellite launch vehice, European Space Agency, maiden flight on June 4, 1996, 40 seconds after sw error boosters and main rocket engine swivel to extreme positions, sharply off-course, core and booster lost contact, vehicle self-destructs itself, satellites carried $500 million were not insured

error in converting 64-bit floating point into 16-bit integer, exceeding maximum storage capacity of integer, no exception handling for this exception leading a crash

Ariane 5

if there are two or more ways to do something,
and one of those...can result in a catastrophe,
then someone will do it.

faulty code comes from Ariane 4, 64-bit floating value to 16-bit integer value conversion for horizontal bias of the launched vehicle, engineers determined that stored value can never be larger than 16-bit integer storage capability, **no need for an error handler for an error that cannot occur** (hey Murphy! :p), code moved as is into Ariane 5 design for reuse, extremely costly mistake, Ariane 5 much faster than Ariane 4 and values larger than 16-bit integer could be quite common, original assumptions didn't hold

--edward a. murphy, jr., 1949

AT&T long distance network

January 15, 1990 AT&T long distance network seriously disrupted, half of computerized telephone-routing switches crashed, remainder hw based switches collapsed under heavy workload, 70 million long-distance calls dismissed, 60000 people lost all telephone service, millions of dollars revenue lost but most importantly the credibility and reputation is ruined

AT&T long distance network

network crash due to single faulty line of code in error-recovery procedure, if server discovers and error state it reboots itself, a crude but effective way of “wiping the state clean”, after reboot switch broadcasts an “OK” msg to other switches to let them know its back online, the error occurs when a very busy switch receives and OK msg, when there is a delay in handling OK msg this condition forces recipient server into error state and reboot... a catastrophe possibility...

AT&T long distance network

15 Jan, 1990, System 7 (sw version) switch in New York City rebooted itself following an error detection, after reboot broadcasts online msg, all OK msgs handled correctly except very busy 3 switches at St.Louis, Detroit and Atlanta, these switches get in an error state and rebooted themselves, when they rebooted they broadcasted their OK msgs over network...

AT&T long distance network

two sided problem,

1) when switch is down, it pushes all of its long-distance traffic to other switches making them busier

2) when switch comes back broadcasted “OK” msg troubles already busy switches again

AT&T long distance network

two sided problem,

1) when switch is down, it pushes all of its long-distance traffic to other switches making them busier

2) when switch comes back broadcasted “OK” msg troubles already busy switches again

some switches started repeatedly rebooting under many OK msgs, within 10 minutes half of the switches in AT&T network failed

AT&T long distance network

crash could have been worse because AT&T only converted 80 of its network switches to System 7 software, they had left 34 System 6 software switches for **“just-in-case”(!!!)** that didn’t crash

robot missions to Mars

$125 million Mars Climate Orbiter to facilitate communications and send probes on Mars, the spacecraft is lost because of miscommunication between two support teams on Earth

robot missions to Mars

$125 million Mars Climate Orbiter to facilitate communications and send probes on Mars, the spacecraft is lost because of miscommunication between two support teams on Earth

Lockheed Martin, 2 teams, Colorado flight operation team using English units (foot-pounds), California navigation team using metric units (Newton), unaware of each others preference, the program requires input in Newtons where 1 Newton = 4.45 foot-pounds

robot missions to Mars

September 23, 1999, Mars Climate Orbiter approaching the Red Planet, firing engines required while orbiting, units mismatched, navigation team specified 4.45 times too much thrust, spacecraft flew low and burned into atmosphere...

robot missions to Mars

September 23, 1999, Mars Climate Orbiter approaching the Red Planet, firing engines required while orbiting, units mismatched, navigation team specified 4.45 times too much thrust, spacecraft flew low and burned into atmosphere...

a few months later Mars Polar Lander, $165 million, supposed to land on south pole of Mars, December 3, 1999, lost contact, engineers suspect that sw got false signal and shutdown engines 100 feet above surface

robot missions to Mars

Tony Spear, project manager of Mars Pathfinder mission declared that:

"It is just as hard to do Mars missions now as it was in the mid-70s. I'm a big believer that software hasn't gone anywhere. Software is the number-one problem."

robot missions to Mars

Tony Spear, project manager of Mars Pathfinder mission declared that:

"It is just as hard to do Mars missions now as it was in the mid-70s. I'm a big believer that software hasn't gone anywhere. Software is the number-one problem."

NASA then successfully landed two rovers consecutively in 2003 and 2004 (Opportunity and Spirit), they greatly exceeded their goals, both rovers were still operational after 19 months

Therac-25

German physicist, Wilhelm Roentgen, 1895, x-ray, 50-60% of cancer patients are treated with radiation today, linear accelerators for electron beams to shallow tumors, x-ray beams to reach deeper tumors

Therac-25

German physicist, Wilhelm Roentgen, 1895, x-ray, 50-60% of cancer patients are treated with radiation today, linear accelerators for electron beams to shallow tumors, x-ray beams to reach deeper tumors

Therac-25 linear accelerator was notoriously unreliable, even not unusual for system malfunction 40 times a day...

Therac-25

German physicist, Wilhelm Roentgen, 1895, x-ray, 50-60% of cancer patients are treated with radiation today, linear accelerators for electron beams to shallow tumors, x-ray beams to reach deeper tumors

Therac-25 linear accelerator was notoriously unreliable, even not unusual for system malfunction 40 times a day...

important example: how the safety of system relies solely upon the quality of its embedded software... causing harm...

Therac-25

20 month period between June 1985 – January 1987, massive overdoses to 6 patients, causing 3 deaths, 1987 may seem like distant past but entire story has been throughly researched and documented...

Therac-25

20 month period between June 1985 – January 1987, massive overdoses to 6 patients, causing 3 deaths, 1987 may seem like distant past but entire story has been throughly researched and documented...

AECL (Canada) and CGR (French) cooperatively built two linear accelerators in 1970s: Therac-6 and Therac-20, modernization of old versions, with computer – ease of operation, **safety features were built into hardware**...

Therac-25

After Therac-20 CGR left issue, AECL continued for next-generation Therac-25, software now an integral part of system, incapable of operating without computer, replacing hw safety features of Therac-20 with software safety features in Therac-25

Therac-25

After Therac-20 CGR left issue, AECL continued for next-generation Therac-25, software now an integral part of system, incapable of operating without computer, replacing hw safety features of Therac-20 with software safety features in Therac-25

reusing code of Therac-6 and Therac-20, saving time and money, “tried and true” sw is more reliable than fresh code (this is what we’re told in some of the textbooks ☺)...

Therac-25

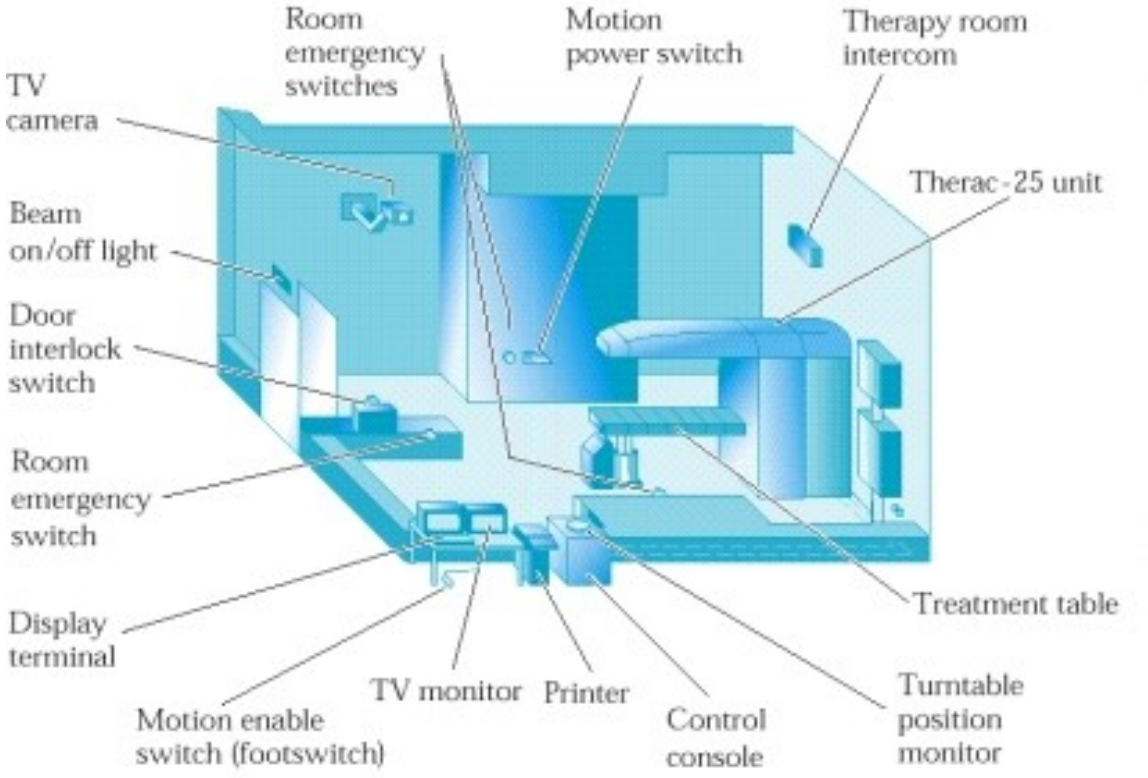
After Therac-20 CGR left issue, AECL continued for next-generation Therac-25, software now an integral part of system, incapable of operating without computer, replacing hw safety features of Therac-20 with software safety features in Therac-25

reusing code of Therac-6 and Therac-20, saving time and money, “tried and true” sw is more reliable than fresh code (this is what we’re told in some of the textbooks ☺)...

first Therac-25 in 1983, 11 systems in Canada and US

Therac-25

large machine in its own room, shielding walls, ceilings, floor from radiation, TV cam, microphone, speaker for communication between technician and patient



Marietta, Georgia, June 1985

61 years old breast cancer patient, radiation to collarbone, she complained that she had been burnt

Marietta, Georgia, June 1985

61 years old breast cancer patient, radiation to collarbone, she complained that she had been burnt

hospital physicist contacted AECL and reported case (maybe a bug in system), AECL engineers replied that it is not possible

Marietta, Georgia, June 1985

61 years old breast cancer patient, radiation to collarbone, she complained that she had been burnt

hospital physicist contacted AECL and reported case (maybe a bug in system), AECL engineers replied that it is not possible

overdose crippled patient, physicist estimates 75-100 times too large overdosing, patient sued AECL and the hospital in October, 1985

Hamilton, Ontario, July 1985

40 year old woman, cervical cancer, operator started treatment, machine shutdown 5 seconds after with error msg, display shows no delivery of radiation to patient yet, the operator typed “P” for proceed, system shut down again with the same error msg (**recall it was not unusual for the machine to malfunction several dozen times a day**), operator “P” for 3 more times, always same result -> “treatment suspend” mode

Hamilton, Ontario, July 1985

40 year old woman, cervical cancer, operator started treatment, machine shutdown 5 seconds after with error msg, display shows no delivery of radiation to patient yet, the operator typed "P" for proceed, system shut down again with the same error msg (**recall it was not unusual for the machine to malfunction several dozen times a day**), operator "P" for 3 more times, always same result -> "treatment suspend" mode

operator went into room, patient complained that she has been burnt, reported AECL, when she returned for further treatment 3 days later it was discovered that she was overdosed for 65-85 times, died in November, 1985

1. AECL investigation, July-September 1985

engineer to investigate, unable to reproduce overdose but uncovering design problems about a microswitch, AECL introduced hw and sw fixes to microswitch problem

Yakima, Washington, December 1985

Woman receiving radiation therapy series developed a strange reddening on her hip (several parallel stripes), staff tries to determine the cause, maybe accelerator's blocking trays which have already been discarded, suspected radiation overdose and contacted AECL by letter and phone

Yakima, Washington, December 1985

Woman receiving radiation therapy series developed a strange reddening on her hip (several parallel stripes), staff tries to determine the cause, maybe accelerator's blocking trays which have already been discarded, suspected radiation overdose and contacted AECL by letter and phone

AECL described that neither Therac-25, nor the operator error could have produced the described damage, they explained in 2 technical pages that why it is impossible for Therac-25 to produce overdose (it also claims that no similar accidents have been reported), patient lived with mild disability

Tyler, Texas, March 1986

male patient for 9th series of radiation treatment for a cancerous tumor on his back, operator enter treatment data into computer, she notices that she typed “X” (for x-ray) instead of “E” (for electron beam), quite common mistake because x-ray treatments are more common, being an experienced operator she fixes her mistake by changing “X” to “E” on screen and moving to the cursor back to “ready” position, system says “beam ready”, she types “B” (beam on), after a few seconds system shut down, “Malfunction 54” -> “treatment pause” a low-priority problem, dose monitor shows patient only received 6 units of dose instead of required 202, operator pushes “P” to proceed...

Tyler, Texas, March 1986

normally there is a videocam and intercom facility for establishing communication in two adjoint rooms, however none were functional at that time

Tyler, Texas, March 1986

normally there is a videocam and intercom facility for establishing communication in two adjoint rooms, however none were functional at that time

patient received 8 prior treatments, he was experienced so he immediately realizes that something is going wrong, he was aware of overdose (later explained that “someone poured hot coffee on his back, or electric shock”)

Tyler, Texas, March 1986

normally there is a videocam and intercom facility for establishing communication in two adjoint rooms, however none were functional at that time

patient received 8 prior treatments, he was experienced so he immediately realizes that something is going wrong, he was aware of overdose (later explained that “someone poured hot coffee on his back, or electric shock”)

when he tries to get up, system delivered second dose hitting him on the arm, pounding on the door warning operator, 80-125 times overdose, losing bodily functions and dying 5 months later

2. AECL investigation, March 1986

after Texas incident, hospital in Texas shutdown Therac-25 system and notified AECL, two engineers to examine system, they told physicians that it was impossible for Therac-25 to overdose a patient and suggested that the patient must have received electroshock due to a leakage in hospital electric system

2. AECL investigation, March 1986

after Texas incident, hospital in Texas shutdown Therac-25 system and notified AECL, two engineers to examine system, they told physicians that it was impossible for Therac-25 to overdose a patient and suggested that the patient must have received electroshock due to a leakage in hospital electric system

hospital checked its electrical system and no fault found, then they double-checked Therac-25 calibration... nothing else to do, they put Therac-25 back into operation

Tyler, Texas, April 1986

replay of prior accident with the same technician, this time intercom working and operator could hear the patient screaming but she couldn’t do anything after, massive dose of radiation to brain killing the patient within 3 weeks

hospital immediately shut off Therac-25 again and contacted AECL

Yakima, Washington, January 1987

second patient here burned with the same of December 1985 accident, 4 days after treatment -> patient's skin revealed a series of parallel red stripes

this time hospital staff matched two cases and perked... patient died three months later

February, 1987

Therac-25 finally declared defective by FDA losing its FDA approval, in order to get approval back AECL must show how they will make system safe

February, 1987

Therac-25 finally declared defective by FDA losing its FDA approval, in order to get approval back AECL must show how they will make system safe

5 months later and after 5 revisions AECL demonstrated a corrective action plan including a variety of hardware interlocks to prevent machine from delivering overdose, or activating the beam when the turntable was not in correct position

software errors

race condition -> two or more tasks sharing a variable –> **extremely difficult to identify and fix**, usually the tasks do not interfere with each other and nothing goes wrong, only in rare conditions...

command screen error

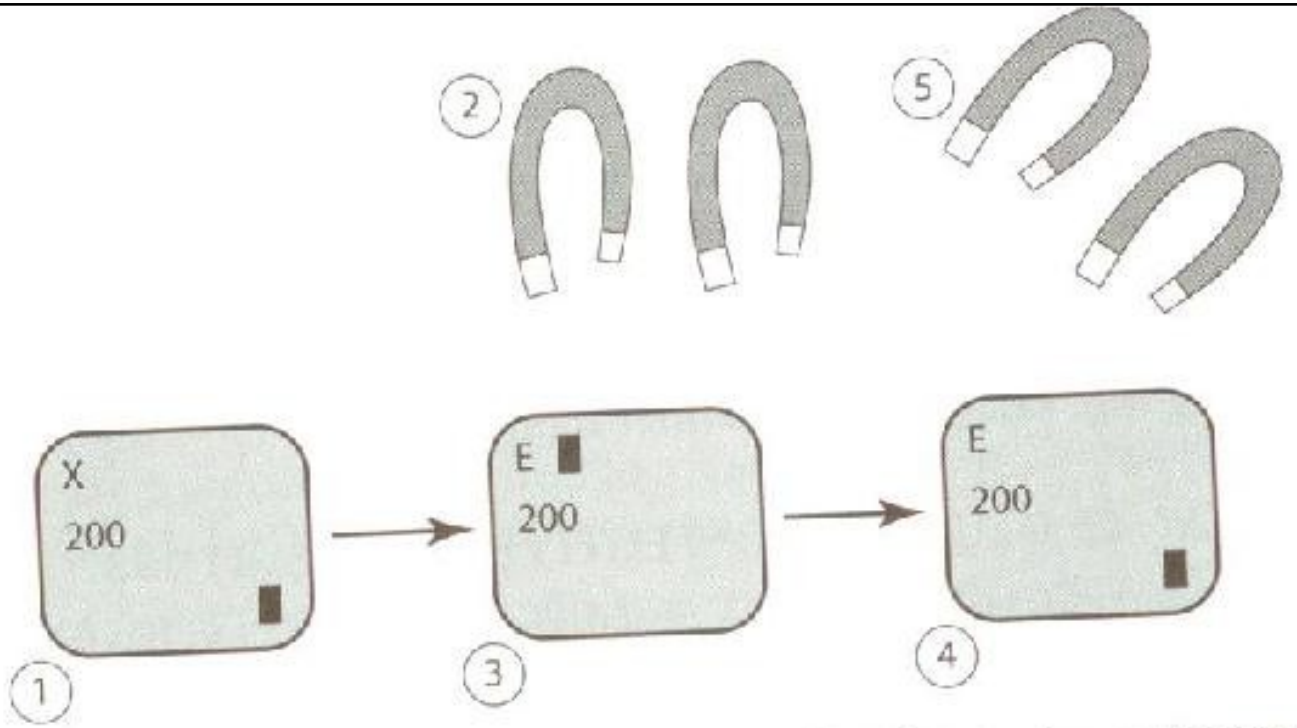


FIGURE 7.4 Illustration of a Therac-25 bug revealed by fast-typing operators. (1) The operator finishes filling in the form. The software knows the form is filled in because the cursor is in the lower right-hand corner of the screen. (2) The software instructs the magnets to move into the correct positions. While the magnets are moving, the software does not check for screen edits. (3) The operator changes the prescription from x-ray to electron beam. (4) The operator finishes the edit, returning the cursor to the lower right-hand corner of the screen. (5) The magnets finish moving. The software now checks the screen cursor. Since it is in the lower right-hand corner, the program assumes there have been no edits.

gun positioning error

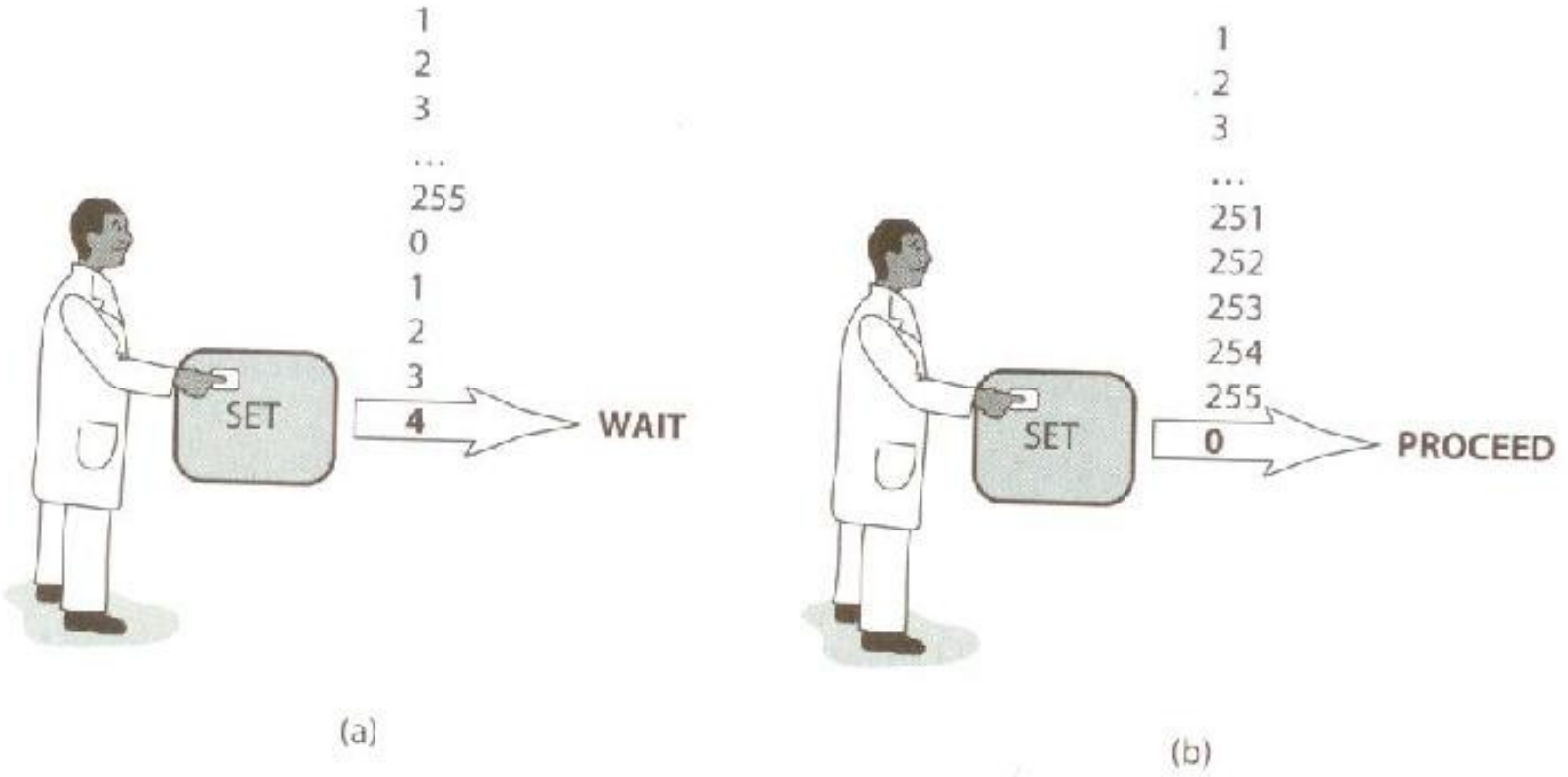


FIGURE 7.5 The Therac-25 could administer radiation too soon if the operator hit the SET button at precisely the wrong time. As long as the electron-beam gun was out of position, a software task kept incrementing an 8-bit variable. (a) Usually when the operator hit the SET button, the variable was not zero and the system would wait, just as it was supposed to. (b) If the operator hit the SET button just at the variable "rolled over" from 255 to 0, the system would administer radiation, even though the gun was out of position.

postmortem

AECL focused on identifying and fixing sw bugs, in fact most accidents are system accidents -> the entire system was broken, not just the software

postmortem

AECL focused on identifying and fixing sw bugs, in fact most accidents are system accidents -> the entire system was broken, not just the software

good engineering:

"A system should be designed so that no single point of failure will lead to a catastrophe."

Therac-25 designers ignored this fundamental engineering principle

postmortem

lack of hw and sw devices to detect and report overdoses and shut down accelerator immediately... instead designers left it up to patients to report when they had received overdoses

postmortem

1) very difficult to find sw bugs in prgs with multiple tasks at the same time and interact through shared variables

postmortem

1) very difficult to find sw bugs in prgs with multiple tasks at the same time and interact through shared variables

2) sw design needs to be **as simple as possible**

postmortem

1) very difficult to find sw bugs in prgs with multiple tasks at the same time and interact through shared variables

2) sw design needs to be **as simple as possible**

3) the code must be reasonably documented **at the time it is written**

postmortem

1) very difficult to find sw bugs in prgs with multiple tasks at the same time and interact through shared variables

2) sw design needs to be **as simple as possible**

3) the code must be reasonably documented **at the time it is written**

4) reuse does **not always** increase the quality of the final product

(earlier codes no problems, because there were hw interlocks, in fact code was faulty)

postmortem

tragedy multiplied because AECL didn't fully communicate with their customers (overdosing impossible :p) even though they had already been sued by the first harmed patient in Georgia

moral responsibility of Therac-25 team

for moral responsibility:

- causal condition: actions (or inactions) of the agent must have caused harm

moral responsibility of Therac-25 team

for moral responsibility:

- causal condition: actions (or inactions) of the agent must have caused harm
- mental condition: actions (or inactions) must have been intended or willed by the agent

moral responsibility of Therac-25 team

for moral responsibility:

- causal condition: actions (or inactions) of the agent must have caused harm
- mental condition: actions (or inactions) must have been intended or willed by the agent

1 absolutely holds but what about 2?

moral responsibility of Therac-25 team

for moral responsibility:

- causal condition: actions (or inactions) of the agent must have caused harm
- mental condition: actions (or inactions) must have been intended or willed by the agent

1 absolutely holds but what about 2? of course didn’t intend but philosophers extend 2 with -> carelessness, recklessness and negligence...

moral responsibility of Therac-25 team

for moral responsibility:

- causal condition: actions (or inactions) of the agent must have caused harm
- mental condition: actions (or inactions) must have been intended or willed by the agent

1 absolutely holds but what about 2? of course didn't intend but philosophers extend 2 with -> carelessness, recklessness and negligence... **Therac-25 team is morally responsible for the deaths.**

Social Impact of Information System Failures, 2009

by Tetsuo TAMAI

computer-based systems, silence when nothing goes wrong, loud outcry when trouble occurs

Social Impact of Information System Failures, 2009

by Tetsuo TAMAI

computer-based systems, silence when nothing goes wrong, loud outcry when trouble occurs

this gives the general public the wrong impression that computer systems are highly unreliable

Social Impact of Information System Failures, 2009

by Tetsuo TAMAI

computer-based systems, silence when nothing goes wrong, loud outcry when trouble occurs

this gives the general public the wrong impression that computer systems are highly unreliable

software is invisible and not easy for ordinary people to understand

computer simulations

computer systems and software behind locked rooms can also cause harm (rather than embedded systems) resulting in poorly designed products, mediocre science, bad policy decisions

computer simulations

computer systems and software behind locked rooms can also cause harm (rather than embedded systems) resulting in poorly designed products, mediocre science, bad policy decisions

computer simulations have a a key role in science and engineering (limits on physcial experiments and applications e.g. too expensive, too time-consuming, unethical, impossible...)

computer simulations

computer systems and software behind locked rooms can also cause harm (rather than embedded systems) resulting in poorly designed products, mediocre science, bad policy decisions

computer simulations have a a key role in science and engineering (limits on physcial experiments and applications e.g. too expensive, too time-consuming, unethical, impossible...)

(used in nuclear weapons, oil search, creating pharmaceuticals, designing better cars...)

uses of simulations

1) modeling past events (e.g. astrophysicists, theories on evolution of universe etc.)

uses of simulations

1) modeling past events (e.g. astrophysicists, theories on evolution of universe etc.)

2) understanding the world around us (geologists, exploration for oil search is quite expensive, networks of microphones and setting off explosives, then graphical representations of rock formations)

uses of simulations

1) modeling past events (e.g. astrophysicists, theories on evolution of universe etc.)

2) understanding the world around us (geologists, exploration for oil search is quite expensive, networks of microphones and setting off explosives, then graphical representations of rock formations)

3) predicting future (weather predictions, economical predictions, soccer predictions ☺ etc.)

uses of simulations

1) modeling past events (e.g. astrophysicists, theories on evolution of universe etc.)

2) understanding the world around us (geologists, exploration for oil search is quite expensive, networks of microphones and setting off explosives, then graphical representations of rock formations)

3) predicting future (weather predictions, economical predictions, soccer predictions ☺ etc.)

simulations may lead to erroneous results

validating simulations

they’re software -> validation and verification

validating simulations

they’re software -> validation and verification

verification -> does the program correctly implement the model? *(are we doing the product right?)*

validating simulations

they’re software -> validation and verification

verification -> does the program correctly implement the model? *(are we doing the product right?)*

validation -> does the model accurately represent the real system? *(are we doing the right product?)*

validating simulations

they’re software -> validation and verification

verification -> does the program correctly implement the model? *(are we doing the product right?)*

validation -> does the model accurately represent the real system? *(are we doing the right product?)*

validating -> duplicating the performance of actual system e.g. crashing real cars, comparing model predictions (for car safety)

validating simulations

validating a model that predicts future can introduce new difficulties

validating simulations

validating a model that predicts future can introduce new difficulties

tomorrow's weather forecast – compare with real value -- wait for tomorrow ☺

validating simulations

validating a model that predicts future can introduce new difficulties

tomorrow's weather forecast – compare with real value -- wait for tomorrow ☺

global warming model for +50 years – you can't wait for 50 years... instead "predict the present" (suppose 75 years data, target predict +25 years, use only 50 years data for model, the rest 25 years data is used for validation)

validating simulations

validating a model that predicts future can introduce new difficulties

tomorrow's weather forecast – compare with real value -- wait for tomorrow ☺

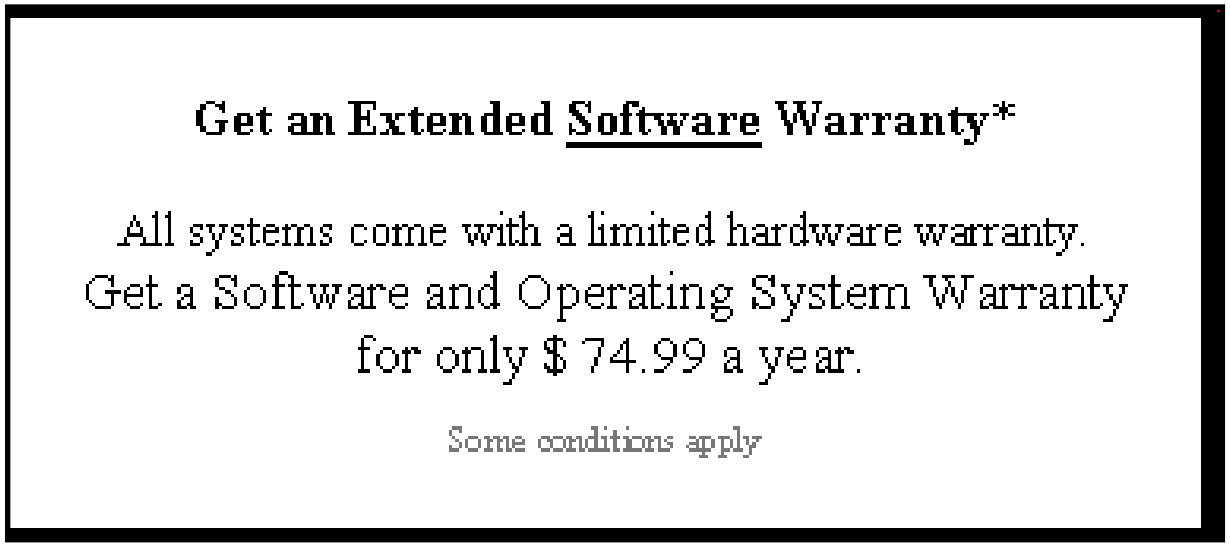
global warming model for +50 years – you can't wait for 50 years... instead "predict the present" (suppose 75 years data, target predict +25 years, use only 50 years data for model, the rest 25 years data is used for validation)

+ credibility check by experts and decision makers

software warranties

Leveson and Turner -> “There’s always another software bug.”

If perfect software is impossible, what kind of warranty should a consumer expect to get from a software company?



shrinkwrap warranties

consumer sw usually counted as shrinkwrap sw (because of plastic wrap ☺) and not too many years ago manufacturers provide no warranty at all

shrinkwrap warranties

consumer sw usually counted as shrinkwrap sw (because of plastic wrap ☺) and not too many years ago manufacturers provide no warranty at all

today, many sw manufacturers (including MS provide 90-day replacement or money-back guarantee if the program “fails”)

shrinkwrap warranties

consumer sw usually counted as shrinkwrap sw (because of plastic wrap ☺) and not too many years ago manufacturers provide no warranty at all

today, many sw manufacturers (including MS provide 90-day replacement or money-back guarantee if the program “fails”)

e.g. at least MS Office states that the sw will do more or less what’s written in its documentation

shrinkwrap warranties

consumer sw usually counted as shrinkwrap sw (because of plastic wrap ☺) and not too many years ago manufacturers provide no warranty at all

today, many sw manufacturers (including MS provide 90-day replacement or money-back guarantee if the program “fails”)

e.g. at least MS Office states that the sw will do more or less what’s written in its documentation

e.g. Railroad Tycoon game -> broken storage medium or installation :p

shrinkwrap warranties

this is “limited warranty”: most vendors may be willing to give you a refund if you cannot get their sw to be installed in your device...

shrinkwrap warranties

this is “limited warranty”: most vendors may be willing to give you a refund if you cannot get their sw to be installed in your device

they of course don’t accept any liability for your business gets harmed because of crashes etc ☺

shrinkwrap warranties

this is “limited warranty”: most vendors may be willing to give you a refund if you cannot get their sw to be installed in your device

they of course don’t accept any liability for your business gets harmed because of crashes etc ☺

In other words:

“Don’t blame us if the program doesn’t do what you hoped it would do, or if it crashes all the time, or if it’s full of bugs” O_o???

are software warranties enforcable?

in US, you cannot put an “unfair warranty” for a product that costs more than $25

if you consider a computer program as a product, hence unfair warranties on shrinkwrap software could be a violation of law

many trials held, but different laws in different countries of course

moral responsibilities of software manufacturers

consider sw manufacturers of shrinkwrapped sw liable for damages, damages caused by errors encountered by licenses

moral responsibilities of software manufacturers

consider sw manufacturers of shrinkwrapped sw liable for damages, damages caused by errors encountered by licenses

manufacturers rely on customers for fixing bugs today -> if responsible, huge test teams (sofware testers) -> higher prices and longer program development periods

moral responsibilities of software manufacturers

consider sw manufacturers of shrinkwrapped sw liable for damages, damages caused by errors encountered by licenses

manufacturers rely on customers for fixing bugs today -> if responsible, huge test teams (sofware testers) -> higher prices and longer program development periods

+ insurance to protect them from lawsuits, usually very expensive and this cost will also be shared by customers

moral responsibilities of software manufacturers

small start-up sofware companies in this sw industry will be much more affected then large-established corporations, slowing the entry of new compaines in field, decreasing the level of innovation and vitality in sw industry

moral responsibilities of software manufacturers

small start-up sofware companies in this sw industry will be much more affected then large-established corporations, slowing the entry of new compaines in field, decreasing the level of innovation and vitality in sw industry

consumers will have confidence in sw they bought, fewer products, very high prices but much more reliable

moral responsibilities of software manufacturers

small start-up sofware companies in this sw industry will be much more affected then large-established corporations, slowing the entry of new compaines in field, decreasing the level of innovation and vitality in sw industry

consumers will have confidence in sw they bought, fewer products, very high prices but much more reliable

utilitarian analysis may conclude for current scheme

moral responsibilities of software manufacturers

however consider this hypothetical scenario:

consumer buys a game named “Incredible Bulk” for $49,95

moral responsibilities of software manufacturers

however consider this hypothetical scenario:

consumer buys a game named “Incredible Bulk” for $49,95

the game is playable but have annoying bugs, next year comes “Incredible Bulk II”

moral responsibilities of software manufacturers

however consider this hypothetical scenario:

consumer buys a game named “Incredible Bulk” for $49,95

the game is playable but have annoying bugs, next year comes “Incredible Bulk II”

consumer wants to see the bugs fixed and buys the new version, there are also new features -> resulting in new bugs

moral responsibilities of software manufacturers

however consider this hypothetical scenario:

consumer buys a game named “Incredible Bulk” for $49,95

the game is playable but have annoying bugs, next year comes “Incredible Bulk II”

consumer wants to see the bugs fixed and buys the new version, there are also new features -> resulting in new bugs

but don’t worry, next year “Incredible Bulk III” is coming :p

moral responsibilities of software manufacturers

however consider this hypothetical scenario:

consumer buys a game named “Incredible Bulk” for $49,95

the game is playable but have annoying bugs, next year comes “Incredible Bulk II”

consumer wants to see the bugs fixed and buys the new version, there are also new features -> resulting in new bugs

but don’t worry, next year “Incredible Bulk III” is coming :p **Is this a fair arrangement?**

moral responsibilities of software manufacturers

from SCT view it’s unfair

right to know about bugs, open-fair contract, manufacturer should be open about disclosing weaknesses in its product

moral responsibilities of software manufacturers

from SCT view it’s unfair

right to know about bugs, open-fair contract, manufacturer should be open about disclosing weaknesses in its product

consumer organizations test products for potential buyers by publishing reviews

moral responsibilities of software manufacturers

from SCT view it's unfair

right to know about bugs, open-fair contract, manufacturer should be open about disclosing weaknesses in its product

consumer organizations test products for potential buyers by publishing reviews

consumer purchases the right to use product A manufacturer has to remove defects without enforcing purchase of additional features, patches free on web

moral responsibilities of software manufacturers

from SCT view it's unfair

right to know about bugs, open-fair contract, manufacturer should be open about disclosing weaknesses in its product

consumer organizations test products for potential buyers by publishing reviews

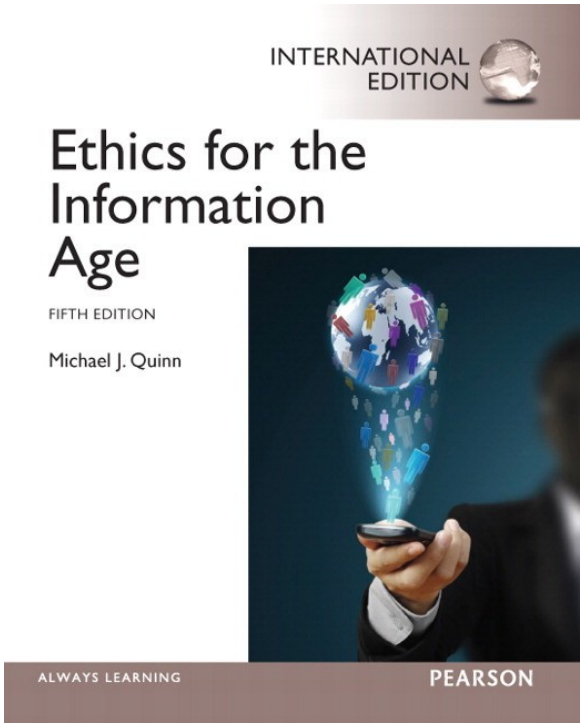
consumer purchases the right to use product A manufacturer has to remove defects without enforcing purchase of additional features, patches free on web, **witholding patches until next major release is wrong**

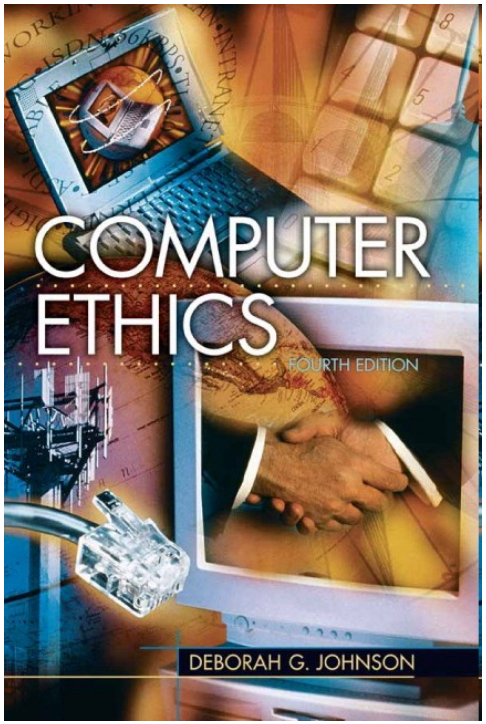
references

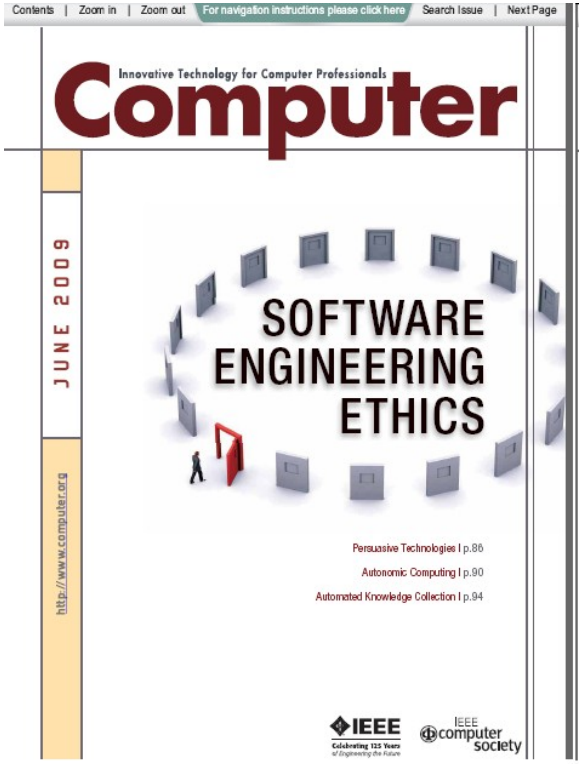
References

"Patriot Missile Defense – Software Problem Led to System Failure at Dhahran, Saudi Arabia", GAO Report, General Accounting Office, Washington DC, February 4, 1992.

INTRODUCTION, APPROXIMATION AND ERRORS	
Topic	Sources of error
Summary	Textbook notes on sources of error
Major	General Engineering
Authors	Autar Kaw
Date	April 24, 2009
Web Site	http://numericalmethods.eng.usf.edu







Google™

WIKIPEDIA