



Chapter 7

Computer Reliability



Chapter Overview

- Introduction
- Data-entry or data-retrieval errors
- Software and billing errors
- Notable software system failures
- Therac-25
- Computer simulations
- Software engineering
- Software warranties

Copyright © 2008 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4-2

Introduction

- Computer systems are sometimes unreliable
 - Erroneous information in databases
 - Misinterpretation of database information
 - Malfunction of embedded systems
- Effects of computer errors
 - Inconvenience
 - Bad business decisions
 - Fatalities

Copyright © 2008 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4-3

Data-Entry or Data-Retrieval Errors

- Disfranchised voters
- False arrests
- Analysis: Accuracy of NCIC records

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4-4**

Disfranchised Voters

- November 2000 general election
- Florida disqualified thousands of voters
- Reason: People identified as felons
- Cause: Incorrect records in voter database
- Consequence: May have affected election's outcome

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4-5**

False Arrests

- Sheila Jackson Stossier mistaken for Shirley Jackson
 - Arrested and spent five days in detention
- Roberto Hernandez mistaken for another Roberto Hernandez
 - Arrested twice and spent 12 days in jail
- Terry Dean Rogan arrested after someone stole his identity
 - Arrested five times, three times at gun point

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4-6**

Accuracy of NCIC Records

- March 2003: Justice Dept. announces FBI not responsible for accuracy of NCIC information
- Exempts NCIC from some provisions of Privacy Act of 1974
- Should government take responsibility for data correctness?

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4-7**

Dept. of Justice Position

- Impractical for FBI to be responsible for data's accuracy
- Much information provided by other law enforcement and intelligence agencies
- Agents should be able to use discretion
- If provisions of Privacy Act strictly followed, much less information would be in NCIC
- Result: fewer arrests

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4-8**

Position of Privacy Advocates

- Number of records is increasing
- More erroneous records → more false arrests
- Accuracy of NCIC records more important than ever

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4-9**

Analysis: Database of Stolen Vehicles

- > 1 million cars stolen every year
 - Owners suffer emotional, financial harm
 - Raises insurance rates for all
- Transporting stolen car across a state line
 - Before NCIC, greatly reduced chance of recovery
 - After NCIC, nationwide stolen car retrieval
- At least 50,000 recoveries annually due to NCIC
- Few stories of faulty information causing false arrests
- Benefit > harm → Creating database the right action

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 10

Software and Billing Errors

- Errors leading to system malfunctions
- Errors leading to system failures
- Analysis: E-retailer posts wrong price, refuses to deliver

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 11

Errors Leading to System Malfunctions

- Qwest sends incorrect bills to cell phone customers
- Faulty USDA beef price reports
- U.S. Postal Service returns mail addressed to Patent and Trademark Office
- Spelling and grammar error checkers increased errors
- BMW on-board computer failure

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 12

Errors Leading to System Failures

- Los Angeles County + USC Medical Center laboratory computer
- Japan's air traffic control system
- Chicago Board of Trade
- London International Financial Futures and Options Exchange
- Comair's Christmas Day shutdown

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 13

Analysis: E-Retailer Posts Wrong Price, Refuses to Deliver

- Amazon.com in Britain offered iPod for 7 pounds instead of 275 pounds
- Orders flooded in
- Amazon.com shut down site, refused to deliver unless customers paid true price
- Was Amazon.com wrong to refuse to fill the orders?

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 14

Rule Utilitarian Analysis

- Imagine rule: A company must always honor the advertised price
- Consequences
 - More time spent proofreading advertisements
 - Companies would take out insurance policies
 - Higher costs → higher prices
 - All consumers would pay higher prices
 - Few customers would benefit from errors
- Conclusion
 - Rule has more harms than benefits
 - Amazon.com did the right thing

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 15

Kantian Analysis

- Buyers knew 97.5% markdown was an error
- They attempted to take advantage of Amazon.com's stockholders
- They were not acting in "good faith"
- Buyers did something wrong

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 16**

Notable Software System Failures

- Patriot Missile
- Ariane 5
- AT&T long-distance network
- Robot missions to Mars
- Denver International Airport

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 17**

Patriot Missile

- Designed as anti-aircraft missile
- Used in 1991 Gulf War to intercept Scud missiles
- One battery failed to shoot at Scud that killed 28 soldiers
- Designed to operate only a few hours at a time
- Kept in operation > 100 hours
- Tiny truncation errors added up
- Clock error of 0.3433 seconds → tracking error of 687 meters

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 18**

Ariane 5

- Satellite launch vehicle
- 40 seconds into maiden flight, rocket self-destructed
 - \$500 million of uninsured satellites lost
- Statement assigning floating-point value to integer raised exception
- Exception not caught and computer crashed
- Code reused from Ariane 4
 - Slower rocket
 - Smaller values being manipulated
 - Exception was impossible

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 19

AT&T Long-Distance Network

- Significant service disruption
 - About half of telephone-routing switches crashed
 - 70 million calls not put through
 - 60,000 people lost all service
 - AT&T lost revenue and credibility
- Cause
 - Single line of code in error-recovery procedure
 - Most switches running same software
 - Crashes propagated through switching network

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 20

Robot Missions to Mars

- Mars Climate Orbiter
 - Disintegrated in Martian atmosphere
 - Lockheed Martin design used English units
 - Jet Propulsion Lab design used metric units
- Mars Polar Lander
 - Crashed into Martian surface
 - Engines shut off too soon
 - False signal from landing gear

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 21

Denver International Airport

- BAE attempted an automated baggage handling system
 - using a system of conveyors and carts which would deliver individual bags to specified destinations.

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 22**

Proposed System

- over 17 miles of track
- 5.5 miles of conveyors
- 4000 carts
- 2700 photo cells
- 59 bar code readers
- 311 radio frequency readers
- more than 150 computers

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 23**

Consequence

- Problems
 - Airport designed before automated system chosen
 - Timeline too short
 - System complexity exceeded development team's ability
- Results
 - Added conventional baggage system
 - 16-month delay in opening airport
 - Cost Denver \$1 million a day

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 24**

Therac-25

- Genesis of the Therac-25
- Chronology of accidents and AECL responses
- Software errors
- Post mortem
- Moral responsibility of the Therac-25 team

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 25**

Genesis of the Therac-25

- AECL and CGR built Therac-6 and Therac-20
- Therac-25 built by AECL
 - PDP-11 an integral part of system
 - Hardware safety features replaced with software
 - Reused code from Therac-6 and Therac-20
- First Therac-25 shipped in 1983
 - Patient in one room
 - Technician in adjoining room

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 26**

Chronology of Accidents and AECL Responses

- Marietta, Georgia (June 1985)
- Hamilton, Ontario (July 1985)
- First AECL investigation (July-Sept. 1985)
- Yakima, Washington (December 1985)
- Tyler, Texas (March 1986)
- Second AECL investigation (March 1986)
- Tyler, Texas (April 1986)
- Yakima, Washington (January 1987)
- FDA declares Therac-25 defective (February 1987)

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 27**

Software Errors

- Race condition: order in which two or more concurrent tasks access a shared variable can affect program's behavior
- Two race conditions in Therac-25 software
 - Command screen editing
 - Movement of electron beam gun

Copyright © 2008 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 28**

Post Mortem

- AECL focused on fixing individual bugs
- System not designed to be fail-safe
- No devices to report overdoses
- Software lessons
 - Difficult to debug programs with concurrent tasks
 - Design must be as simple as possible
 - Documentation crucial
 - Code reuse does not always lead to higher quality
- AECL did not communicate fully with customers

Copyright © 2008 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 29**

Moral Responsibility of the Therac-25 Team

- Conditions for moral responsibility
 - Causal condition: actions (or inactions) caused the harm
 - Mental condition
 - Actions (or inactions) intended or willed -OR-
 - Moral agent is careless, reckless, or negligent
- Therac-25 team morally responsible
 - They constructed the device that caused the harm
 - They were negligent

Copyright © 2008 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 30**

Computer Simulations

- Uses of simulation
- Validating simulations

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 31**

Uses of Simulations

- Simulations replace physical experiments
 - Experiment too expensive or time-consuming
 - Experiment unethical
 - Experiment impossible
- Model past events
- Understand world around us
- Predict the future

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 32**

Validating Simulations

- Verification: Does program correctly implement model?
- Validation: Does the model accurately represent the real system?
- Validation methods
 - Make prediction, wait to see if it comes true
 - Predict the present from old data
 - Test credibility with experts and decision makers

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 33**

Software Engineering

- Specification
- Development
- Validation (testing)
- Software quality is improving

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 34**

Specification

- Determine system requirements
- Understand constraints
- Determine feasibility
- End products
 - High-level statement of requirements
 - Mock-up of user interface
 - Low-level requirements statement

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 35**

Development

- Create high-level design
- Discover and resolve mistakes, omissions in specification
- CASE tools to support design process
- Object-oriented systems have advantages
- After detailed design, actual programs written
- Result: working software system

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 36**

Validation (Testing)

- Ensure software satisfies specification
- Ensure software meets user's needs
- Challenges to testing software
 - Noncontinuous responses to changes in input
 - Exhaustive testing impossible
 - Testing reveals bugs, but cannot prove none exist
- Test modules, then subsystems, then system

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley Slide 4- 37

Software Quality Is Improving

- Standish Group tracks IT projects
- Situation in 1994
 - 1/3 projects cancelled before completion
 - 1/2 projects had time and/or cost overruns
 - 1/6 projects completed on time / on budget
- Situation in 2002
 - 1/6 projects cancelled
 - 1/2 projects had time and/or cost overruns
 - 1/3 projects completed on time / on budget

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley Slide 4- 38

The New York Times
nytimes.com

March 9, 2005

Doctors' Journal Says Computing Is No Panacea

- Modern information technology, many health experts insist, can deliver a huge payoff: fewer medical errors, lower costs and better care.

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley Slide 4- 40

- One paper, based on a lengthy study at a large teaching hospital, found 22 ways that a computer system for physicians could increase the risk of medication errors.
- Role of Computerized Physician Order Entry Systems in Facilitating Medication Errors. JAMA 293(10), March 2005, pp. 1197-1203.

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley Slide 4- 41

- roughly 75% of all large IT projects in health care fail.
- Wears and Berg. Computer Technology and Clinical Work: Still Waiting for Godot. JAMA 293(10), March 2005, pp. 1261-1263.

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley Slide 4- 42

Software Warranties

- Shrinkwrap warranties
- Are software warranties enforceable?
- Uniform Computer Information Transaction Act
- Moral responsibility of software manufacturers

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 43**

Shrinkwrap Warranties

- Some say you accept software “as is”
- Some offer 90-day replacement or money-back guarantee
- None accept liability for harm caused by use of software

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 44**

Are Software Warranties Enforceable?

- Article 2 of Uniform Commercial Code
- Magnuson-Moss Warranty Act
- *Step-Saver Data Systems v. Wyse Technology and The Software Link*
- *ProCD, Inc. v. Zeidenberg*
- *Mortensen v. Timberline Software*

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley **Slide 4- 45**

Uniform Computer Information Transaction Act

- National Conference of Commissioners on Uniform State Laws drafted UCITA
- Under UCITA, software manufacturers can
 - License software
 - Prevent software transfer
 - Disclaim liability
 - Remote disable licensed software
 - Collect information about how software is used
- UCITA applies to software in computers, not embedded systems

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 46

Arguments in Favor of UCITA

- Article 2 of the UCC not appropriate for software
- UCITA recognizes there is no such thing as perfect software
- UCITA prevents software fraud

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 47

Arguments Against UCITA

- Customers should be allowed to purchase software
- UCITA bans giving away software
- UCITA removes software from protections of Magnuson-Moss Act
- UCITA codifies practice of hiding warranty
- UCITA allows "trap doors"
- UCITA restricts free speech
- Fuzzy line between embedded systems & computers
- UCITA is unlikely to pass without amendments

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 48

Moral Responsibility of Software Manufacturers

- If vendors were responsible for harmful consequences of defects
 - Companies would test software more
 - They would purchase liability insurance
 - Software would cost more
 - Start-ups would be affected more than big companies
 - Less innovation in software industry
 - Software would be more reliable
- Making vendors responsible for harmful consequences of defects may be wrong
- Consumers should not have to pay for bug fixes

Copyright © 2006 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Slide 4- 49
