

CSci 780 Advanced Software Engineering

“Software’s Chronic Crisis” by W. Wayt Gibbs

9/3/2003

1

Outline

- The Problem
- A Brief History of SE
- Comparing SE to _E
- Some Progress Being Made
- Formal Methods
- Processes
- Capability Maturity Model
- Discussion

9/3/2003

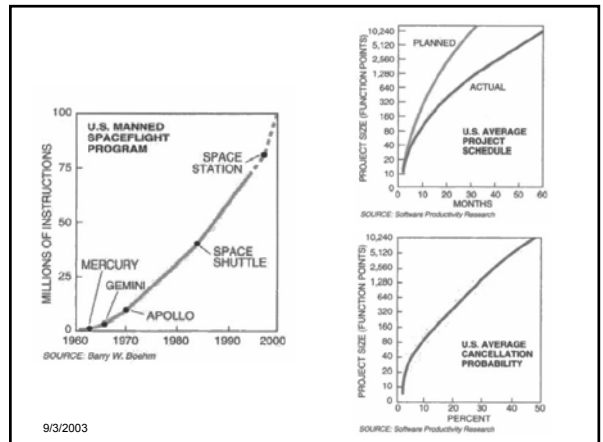
2

The Problem

- 1/4 of large software projects are canceled
- Average software project 50% over cost
- 3/4 of large systems are “operating failures”
- Software in high demand
 - Cell phone: 30 kLOC, 4-speed transmission: 20 kLOC
 - B-2 Stealth Bomber: 3.5 MLOC

9/3/2003

3



9/3/2003

The Denver Airport Case Study

- Big system
 - 21 miles of track, 5.5 miles of conveyors
 - 4,000 luggage cars
 - 5,000 “eyes”, 400 radio rcvrs, 56 bar-code scanners
 - Unique in complexity, technology, capacity
- Software problems
 - Oct 1993 opening slipped to Feb 1995
 - Delays cost about \$500M
 - Had to drastically lower capacity and speeds
 - Lost \$1M per day

9/3/2003

5

Why Did It Fail?

9/3/2003

6

History of SE (1/2)

- 1960's: Programming like using a calculator

As soon as we started programming, we found to our surprise that it wasn't as easy to get programs right as we had thought. Debugging had to be discovered. I can remember the exact instant when I realized that a large part of my life from then on was going to be spent in finding mistakes in my own programs.

- Maurice Wilkes, 1949

9/3/2003

7

History of SE (2/2)

- 1968: NATO Conference
 - Budget and time overruns, faulty software
 - "SE" a provocative aspiration, not a description
- 1978: ICSE, 1989 ISSRE, 1992 FSE
- 1980's: Object-oriented languages
- 1990's: Attempts at component industry, new processes

- But is it "engineering"?

9/3/2003

8

What Are The Hallmarks of Engineering?

9/3/2003

9

But Software Is Different...

- Brooks, "No Silver Bullet"
 - Complexity
 - Invisibility
 - Conformity
 - Changeability
- Discrete, not continuous
- Not subject to physical constraints
- Duplication cost is essentially zero

9/3/2003

10

But Is It Really Different?

- 1628 Swedish ship *Wasa* sinks on maiden voyage due to instabilities in the hull
- 1700's Steam engines repeatedly explode
- 1940 Tacoma Narrows bridge fails due to wind-induced vibrations
- 1997 Mars Pathfinder OS suffers from priority inversion

9/3/2003

11

Compilers & Operating Systems

- Standard architectures
 - Pipeline, microkernel
- Standard texts
 - "Dragon book", "dinosaur book"
- Software Generators
 - LEX, YACC
 - Declarative "4th generation languages"
- Market specialization
 - Edison design group, Microsoft

9/3/2003

12

Causes of Failure

- Shifting requirements
 - Denver had \$20M changes *after construction began*
- New or legacy system dependencies
- Poor specification
- High complexity, coupling
- Large size
- Lack of calendar time
- Insufficient tools and techniques
- Poor management

9/3/2003

13

Some Progress

- Shifting requirements: Prototype-first
- System dependencies: Architectures, processes
- Poor Specification: Formal methods
- High complexity: Domain analysis, architectures
- Large size: Modular decomposition
- Lack of calendar time: Processes
- Insufficient tools and techniques: More work...
- Poor management: Books

9/3/2003

14

Formal Methods

- The science behind software engineering
 - $\oint \vec{E} \cdot d\vec{A} = \frac{q}{\epsilon_0}$ $\oint \vec{B} \cdot d\vec{A} = 0$ $\oint \vec{E} \cdot d\vec{s} = -\frac{d\Phi_B}{dt}$ $\oint \vec{B} \cdot d\vec{s} = \mu_0 i + \frac{1}{c^2} \frac{\partial}{\partial t} \int \vec{E} \cdot d\vec{A}$
 - $\vec{F}_g = m_g \vec{g}$
- Based on sets, Boolean logic, predicate logic
 - Or graph theory, automata theory, probability and statistics

9/3/2003

15

Z: Schema Example

```
[Dog,Alligator]
  Pets
  dogs : F Dog
  alligators : F Alligator
  #dogs > #alligators

my_pets : Pets
my_pets.dogs = {Abbie}
my_pets.alligators = ∅
```

9/3/2003

16

Practical? Cost-Effective?

- Praxis air traffic control project
 - Formal proofs of design properties revealed bugs
 - Finished project on time
- NASA shuttle team, GEC Alstom
- Formal methods still too hard
 - Limited tools, languages
- "I am skeptical that Americans are sufficiently disciplined to apply formal methods in any broad fashion" – David Fisher at NIST

9/3/2003

17

New Processes

- XP: lightweight evolutionary process
 - On-site customer, prototypes
 - Always a working system, trade time for features
 - Write test cases first
- Cleanroom: Don't let bugs in
 - Don't execute code (and maybe don't compile!)
 - Independent verification group
 - Analyze quality statistically
 - Only integrate verified components
- And others...

9/3/2003

18

Evaluation: The Capability Maturity Model

- SEI's Capability Maturity Model (CMM)
- Levels 1 (chaos) to 5 (repeatable, predictable)
- Increased productivity and quality, lower risk
- Understand and fix process problems
- Most organizations are at level 1

9/3/2003

19

Does It Work?

- Raytheon went from level 1 to 2 to 3 '87 to '92
 - 15 projects saved \$15M
 - 2x productivity, 7.7x ROI
- Motorola 1993 report
 - 34 projects assessed at all CMM levels
 - Level 5 vs level 1: defect rate 10x lower, cycle time 8x shorter, productivity 3x better
 - 6.77x ROI
 - Also: no improvement at level 3, costs are high

9/3/2003

20

But...

- Companies can fool the rating
- Discourages companies from hard projects
- Doesn't encourage valuable projects
- CMM is a poor predictor for challenging projects
 - Honeywell (CMM level 5) and QRAS

9/3/2003

21

Discussion

- Should we certify software engineers?
- Is it engineering? Or craft?
- Should we adopt the internship/apprentice model?
- Is the "Eli Whitney" goal realistic?
- What should the relationship between academia and industry be?
- Will the market take care of the problem?

9/3/2003

22

Projects

9/3/2003

23

Some Ideas: Static and Dynamic Analysis

- Do a comparative study of existing static/dynamic analyzers (using your code, or some open source code)
- Implement an analyzer using Valgrind

9/3/2003

24

Some Ideas: Formal Methods

- Write a formal specification of a voting system (or part of your code)
- Write a specification of a protocol in Alloy and analyze it with respect to certain theorems

9/3/2003

25

Some Ideas: Runtime Verification

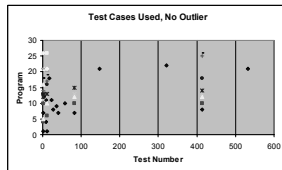
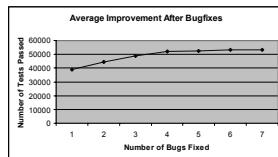
- Write a program using design by contract, ESC Java, etc.
- Embed the spec for DFTs into Nova. Test it exhaustively and see if any assertion is violated.

9/3/2003

26

Some Ideas: Testing

- Do an experiment to compare traditional testing with exhaustive testing



9/3/2003

27