Software Engineering
Lecture 01

Introduction

Referenced documents may be accessed via the URLs located on the course Angel page. Off-campus access will require authentication.

Outline

- What is Software?
- Software Crisis
- What is Software Engineering?
- Why Software Engineering?
- Challenges
- Ethics and Software Engineering

“I think there is a world market for maybe five computers.”

– Thomas J. Watson, Sr., IBM (1949)

“There is no reason anyone would want a computer in their home.”

– Kenneth Olson, DEC (1977)

Dictionary of Computer Quotations, 2nd Ed.

Donald D. Spencer

What is Software?

- Computer programs
- Installation and configuration files
- Documentation
  - System documentation
  - User documentation

- These items are called ***artifacts***

Importance of Software - 1

- Software offers a level of adaptability not possible with many hardware implementations
  - New releases may be distributed at relatively low cost to repair defects or enhance functionality

Importance of Software - 2

"Every business is a software business"

– Watts S. Humphrey

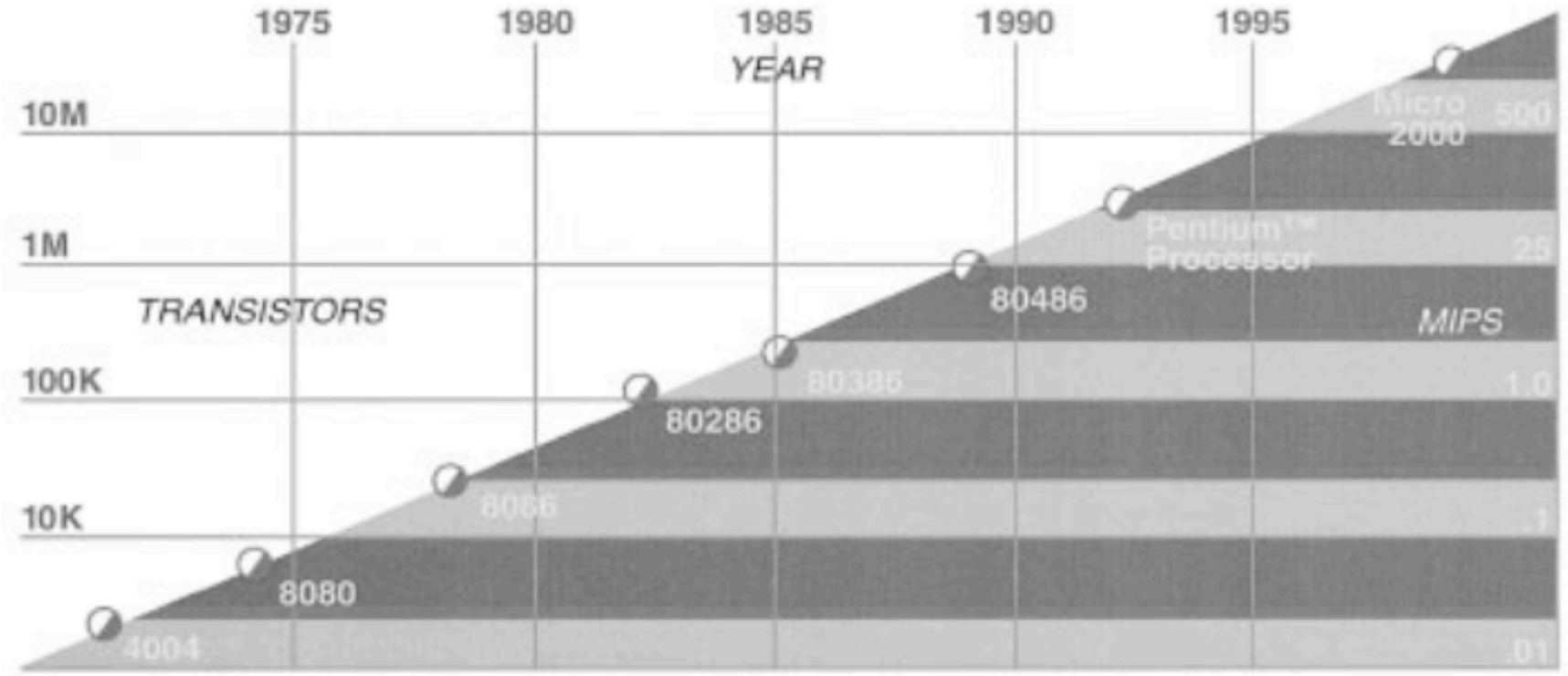
***Winning with Software: An Executive Strategy* (2002)**

Watts S. Humphrey

Software Crisis - 1

- Term coined in the 1960s to refer to the increasing disparities between the capabilities of modern computers and programmers inabilities to use ad-hoc methods to develop increasingly complex software for this hardware while remaining on schedule, within budget, and with acceptable quality

Software Crisis - 2



Moore's Law

The number of transistors in an integrated circuit doubles every two years

Probir K. Bondyopadhyay, “Moore’s Law Governs the Silicon Revolution”,
Proceedings of the IEEE, VOL. 86, NO. 1, JANUARY 1998

Software Project Success Rates

- **Successful** - Completed on time, within budget, & with all specified features
- **Challenged** - Completed but late, over budget, & missing some features
- **Failures** - Cancelled before completion

Category	1994	1996	1998	2000	2004	2009
Successful	16%	27%	26%	28%	29%	32%
Challenged	53%	33%	46%	49%	53%	44%
Failures	31%	40%	28%	23%	18%	24%

Research by Standish Group, www.standishgroup.com

What is Software Engineering?

"The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software."

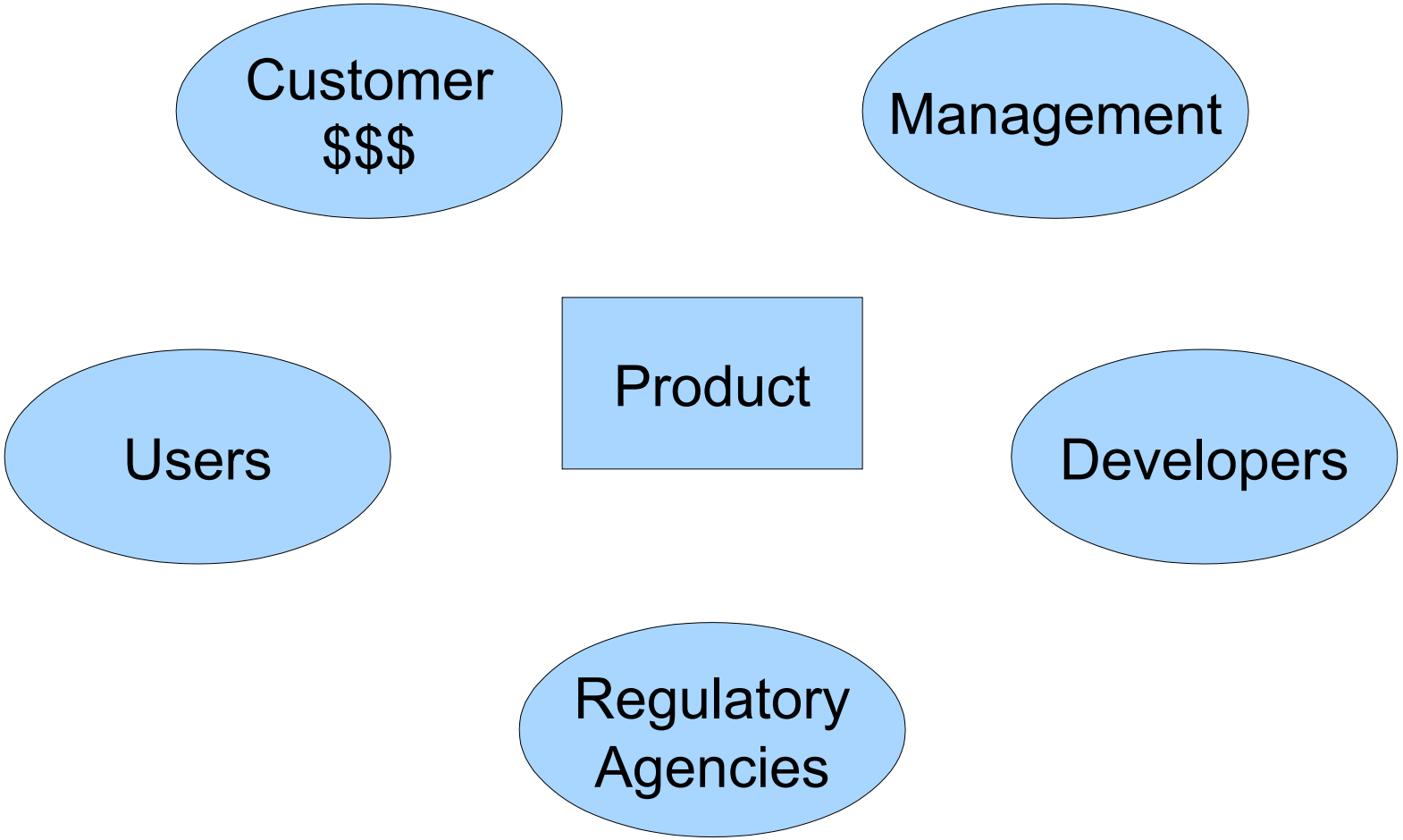
***IEEE Standard Glossary of Software Engineering Terminology*, IEEE Std 610.12-1990**

Challenges – Tradeoffs

- Cost
- Schedule
- Quality

  – “Pick the two you want”

Challenges – Competing Constituencies



Challenges – Inherent Difficulties

- Software is inherently difficult to develop
  - Complex
  - Changeable
  - Difficult to visualize

Frederick P. Brooks, Jr., "No Silver Bullet: Essence and Accidents in Software Engineering", ***IEEE Computer***, vol. 20, no. 2, April 1987, pp. 10-19.

Challenges – Inherent Difficulties

"The hardest single part of building a software system is deciding precisely what to build. No other part of the conceptual work is as difficult as establishing the detailed technical requirements, including all the interfaces to people, machines, and to other software systems."

Frederick P. Brooks, Jr., "No Silver Bullet: Essence and Accidents in Software Engineering", ***IEEE Computer***, vol. 20, no. 2, April 1987, pp. 10-19.

Challenges – Inherent Difficulties

- While coding errors do exist, many errors may be traced back to errors in specification and design
- No silver bullet

Frederick P. Brooks, Jr., "No Silver Bullet: Essence and Accidents in Software Engineering", ***IEEE Computer***, vol. 20, no. 2, April 1987, pp. 10-19.

Cost Impact of Complexity

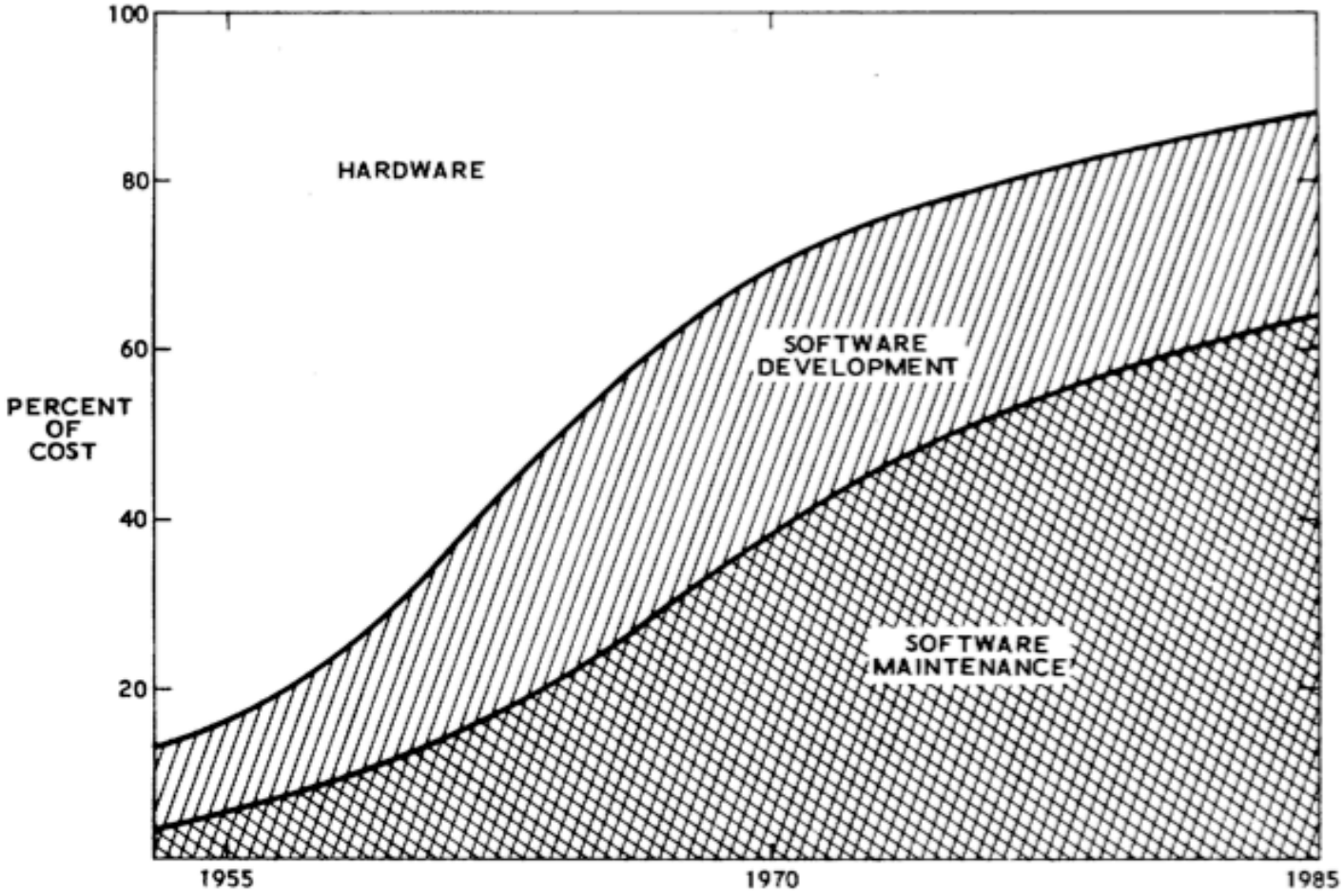
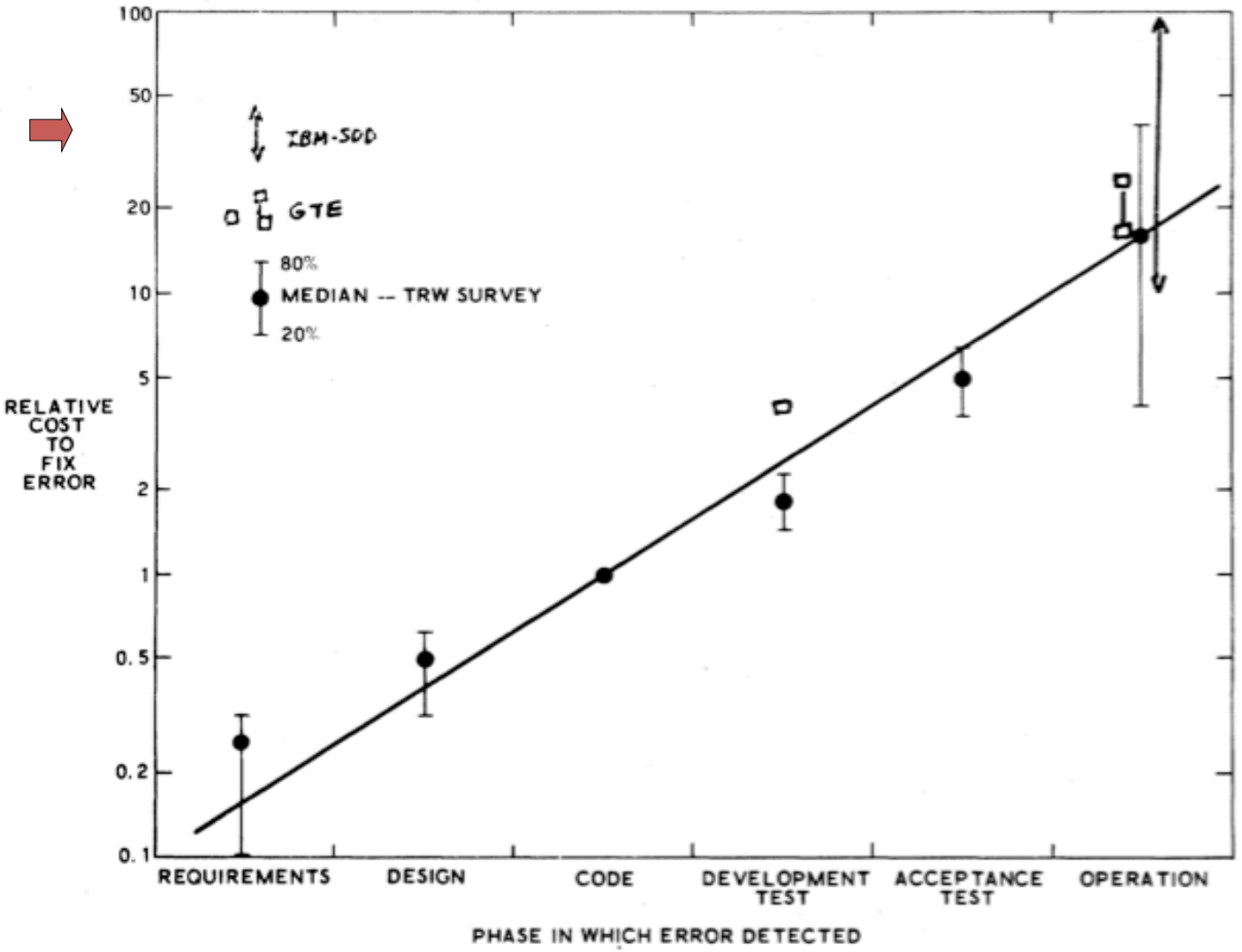


Fig. 1. Hardware–software cost trends.

Barry W. Boehm, "Software Engineering", ***IEEE Transactions on Computers***, vol C-25, no. 12, December 1976, pp. 1226-1241.

Cost Impact of Defects

Logarithmic Scale



Barry W. Boehm, "Software Engineering", ***IEEE Transactions on Computers***, vol C-25, no. 12, December 1976, pp. 1226-1241.

"There is a saying about software quality:

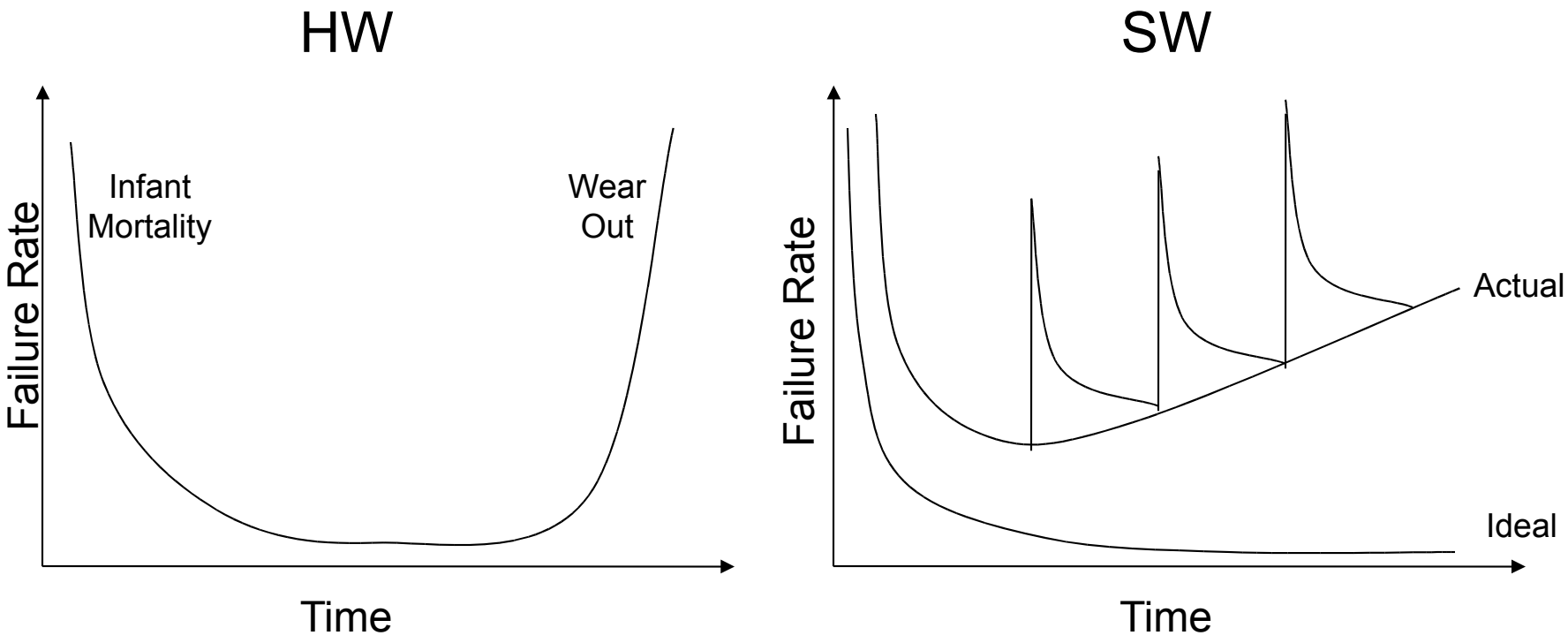
If it doesn't have to work,

we can build it really fast."

Winning with Software: An Executive Strategy

Watts S. Humphrey (2002)

Software Evolves



Roger S. Pressman, ***Software Engineering: A Practitioner's Approach, 5th Ed., 2001***

Ethics and Software Engineering

"The most likely way for the world to be destroyed, most experts agree, is by accident.

That's where we come in; we're computer professionals. We cause accidents. "

-- Nathaniel Borenstein

Source Unknown

Software-Related Accidents

- Therac-25
- Mars Polar Lander [1999]
- Mars Climate Orbiter [1998]
- American Airlines Flight 965 [1995]
- Loss of Ariane 5
- Power-Outage across Northeastern U.S. and Southeastern Canada
- Emergency Shutdown of the Hatch Nuclear Power Plant [2008]

IEEE-CS/ACM
Software Engineering Code of Ethics

1. Public - act consistently with the public interest
2. Client & Employer - act in their best interests and consistent with #1
3. Product - products and modifications should meet highest standards
4. Judgment - maintain integrity and independence
5. Management - ethical approach to SW development and maintenance
6. Profession - advance profession while consistent with #1
7. Colleagues - be fair and supportive
8. Self - lifelong learning and ethical approach to profession

"Any software upgrade costing less than $20 is an admission of guilt."

-- Fred Blechman

Dictionary of Computer Quotations, 2nd Ed.
Donald D. Spencer