04834580 Software Engineering (Honor Track) 2025-26

Introduction to Software Engineering

Sergey Mechtaev
mechtaev@pku.edu.cn
School of Computer Science, Peking University

Software Engineering

Definition (ISO/IEC/IEEE 24765:2017)

The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software. [1]

The Roots of Software Engineering

Margaret Hamilton

Developed on-board flight software for NASA's Apollo program.

> *When I first started using this phrase, it was considered to be quite amusing. It was an ongoing joke for a long time. They liked to kid me about my radical ideas. Software eventually and necessarily gained the same respect as any other discipline. [2]*

1968 NATO Soft. Eng. Conference

1960s saw the emergence "software crisis" — projects were **over budget**, **overdue**, and **unreliable**.

> *The phrase software engineering was deliberately chosen as being provocative, in implying the need for software manufacture to be based on the types of theoretical foundations and practical disciplines that are traditional in the established branches of engineering. [3]*

Catastrophic Impact of Software Bugs

Therac-25: Patients Received Overdoses of Radiation in 1985–87 [4]

Ariane flight v88 failure in 1996 resulted in a loss of US$370 million [5]

Heartbleed affected 17% (around half a million) of web servers in 2014 [6]

Two Boeing 737 MAX crashed in 2018 and 2019 [7]

Software Project High Failure Rate

In 2007, only 30% of government technology-based projects and programmes are successful — Joe Harley, CIO at the UK Department for Work and Pensions. [8]

A 2002–03's survey from 412 experienced (on average, 9 years of experience) project and program managers in the United Kingdom [9]:

- On average, exceed budget by 13%.
- On average, overshot schedule by 20%.
- 9% of projects are abandoned.
- 23% of projects are budget or schedule challenged.

The Mythical Man-Month & No Silver Bullet

Frederick Brooks

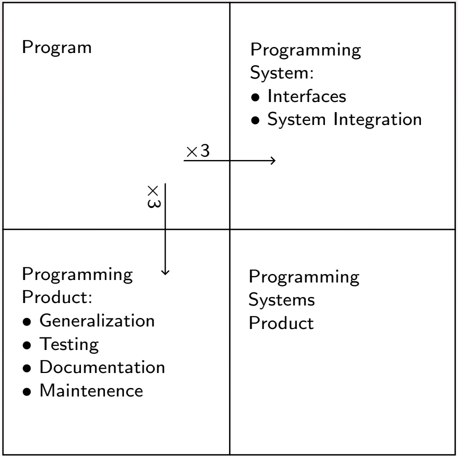
Managed the development of IBM System/360. Turing Award in 1999.

1975	The Mythical Man-Month: Essays on Software Engineering [10]
1989	No Silver Bullet —– Essence and Accident in Software Engineering [11]

Key Ideas

- The cost and effort to move from a program to a system product grow exponentially.
- Adding manpower to a late software project makes it later.
- Larger teams require more channels for communication, which exponentially increases complexity.
- There is no single development, in either technology or management, which by itself promises even one order-of-magnitude improvement.

Brooks: From Program to System Product



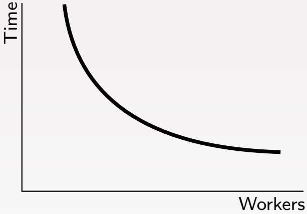
Program	Programming System: ● Interfaces ● System Integration
Programming Product: ● Generalization ● Testing ● Documentation ● Maintenence	Programming Systems Product

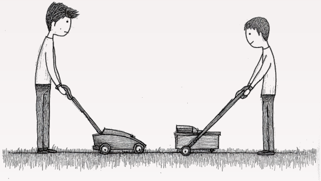
- **Program:** “it works on my machine...”
- **Programming product:** usable in many environments, with various input formats/data, thoroughly tested and documented.
- **Programming system:** interacting components, well-defined interfaces, uses a budget of resources (memory, CPU time), integration testing.
- **Programming systems product:** all of the above.

Brooks' Law

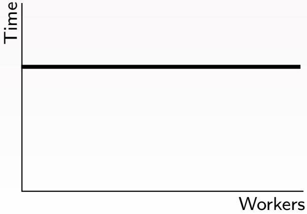
北京大学

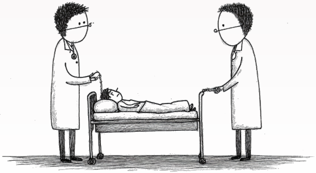
Perfectly partitionable task:





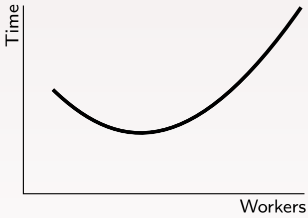
Unpartitionable task:

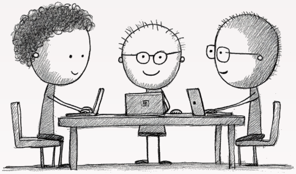




Brooks' Law

Task with complex interrelationships:



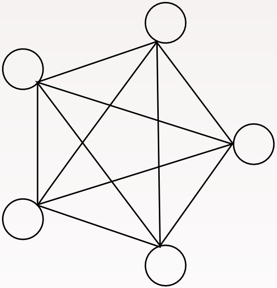


> *Since software construction is inherently a systems effort — and exercise in complex interrelationships — communication effort is great, [...]. Adding more men then lengthens, not shortens, the schedule.*

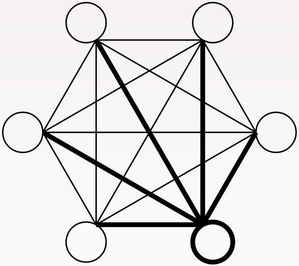
Brooks: Communication Overhead

北京大学

If there are n workers on a project, there are $\frac{n^2-n}{2}$ interfaces across which there may be communication.



5 workers, 10 connections.

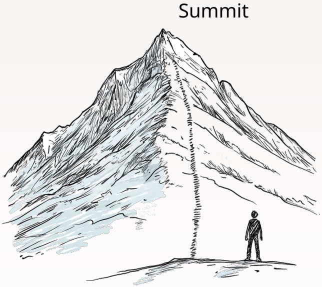


$+1$ worker, $+5$ connections.

Brooks: No Silver Bullet

Essential complexity — the difficulties inherent in the nature of the software:

- Software is complex.
- The complexity is arbitrary (forced without rhyme or reason).
- Software is constantly changing.
- Software is invisible.



Brooks: No Silver Bullet

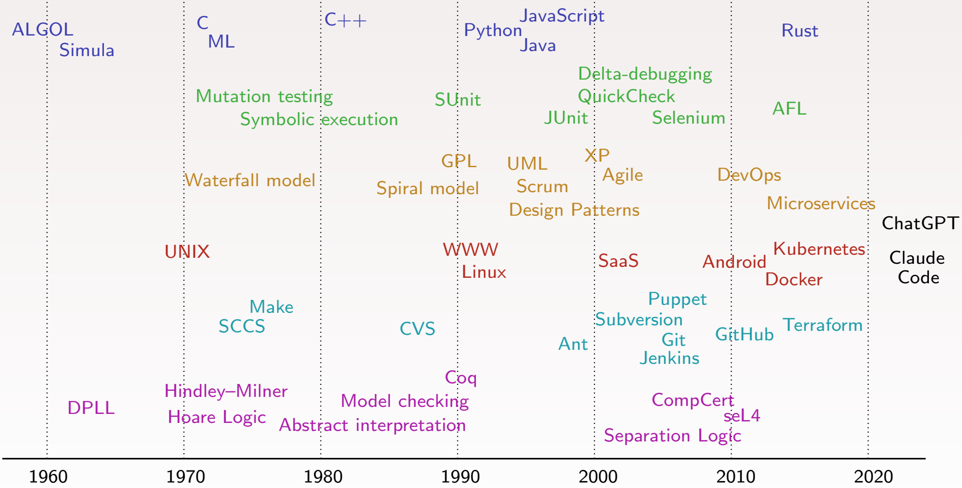
Accidental complexity — the difficulties of today's production, but not inherent.

- Complexity of hardware — addressed by high-level languages.
- Executing multiple programs — addressed by time-sharing, virtualisation.
- Information exchange — addressed by the Internet.

Brooks: No Silver Bullet

There is no single development, in either technology or management technique, which by itself promises even one order of magnitude improvement within a decade in productivity, in reliability, in simplicity.

A Partial Software Engineering Timeline



ALGOL Simula C ML C++ Python JavaScript Java Rust

Mutation testing Symbolic execution SUnit Delta-debugging QuickCheck JUnit Selenium AFL

Waterfall model Spiral model GPL UML Scrum Design Patterns XP Agile DevOps Microservices

ChatGPT Claude Code

UNIX WWW Linux SaaS Android Kubernetes Docker

Make SCCS CVS Ant Puppet Subversion Git Jenkins GitHub Terraform

DPLL Hindley–Milner Hoare Logic Abstract interpretation Model checking Coq CompCert seL4 Separation Logic

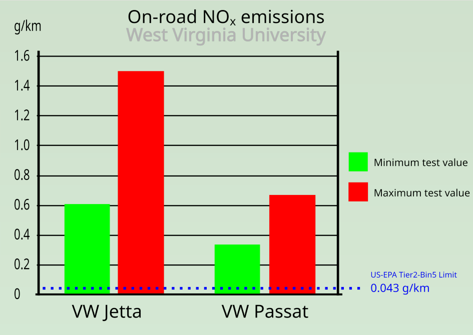
1960 1970 1980 1990 2000 2010 2020

Software Engineering Ethics

Each of us as individual software engineers has an opportunity to make a significant positive impact on society, simply by becoming more sensitive to the long-range human relations implications of our work, and by incorporating this sensitivity into our software designs and products. — Barry Boehm [12]

Dieselgate, Volkswagen Emissions Scandal [13]

In September 2015, the US Environmental Protection Agency charged Volkswagen with installing software in their diesel-powered cars that allowed the vehicles to cheat on EPA emissions tests.



Outcome: over $30 billion in fines and damages — the largest scandal in the history of the automotive industry.

Facebook-Cambridge Analytica Data Privacy Scandal [14] 北京大学

87 million individuals’ data were harvested without their consent to create psychographically tailored advertisements to influence people’s voting preferences in the 2016 US presidential election.

Outcome: Facebook lost more than $60 billion in market capitalization.

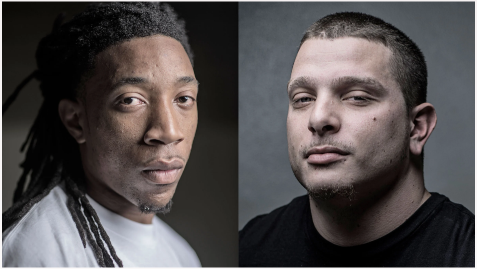
Bias and Fairness in Machine Learning

Definition (Fairness in Decision-Making [15])

The absence of any prejudice or favoritism toward an individual or group based on their inherent or acquired characteristics.

COMPAS, A Discriminatory System [16]

Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) measures the risk of a person to recommit another crime.

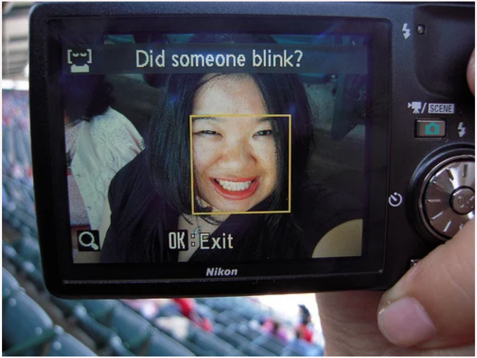


High Risk (10)
Prior Offense: 1 resisting arrest without violence
Subsequent Offenses: None

Low Risk (3)
Prior Offense: 1 attempted burglary
Subsequent Offenses: 3 drug possessions

Nikon Coolpix S630, “Racist Camera” [17]

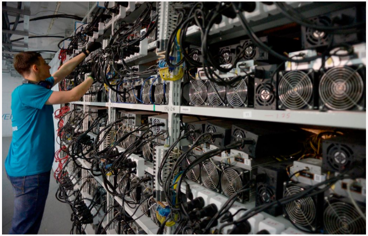
Nikon Coolpix S630 featured “blink warning”.



When Asian faces were photographed, a message would pop up on the camera screen asking, “Did someone blink?”

Bitcoin's Energy Consumption

Proof-of-work blockchains such as Bitcoin are secured and verified by virtual miners. A reward provides an incentive to participate in this type of network. The more computational power one has, the bigger the share of all distributed rewards that go to that miner.



In 2018, Bitcoin's energy consumption is comparable to the one of Ireland [18].

ACM/IEEE-CS Software Engineering Code of Ethics [19]

PUBLIC Software engineers shall act consistently with the public interest.

CLIENT AND EMPLOYER Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.

PRODUCT Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.

JUDGMENT Software engineers shall maintain integrity and independence in their professional judgment.

MANAGEMENT Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.

PROFESSION Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.

COLLEAGUES Software engineers shall be fair to and supportive of their colleagues.

SELF Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.

References I

[1] ISO/IEC/IEEE 24765: Systems and Software Engineering — Vocabulary, 2017.
Available from `https://www.iso.org/standard/71952.html`.

[2] Lori Cameron.
Margaret hamilton: First software engineer.
`https://www.computer.org/publications/tech-news/events/`
`what-to-know-about-the-scientist-who-invented-the-term-software-er`
May 2018.

[3] Brian Randell.
The 1968/69 nato software engineering reports.
History of software engineering, 37, 1996.

[4] Wikipedia contributors.
Therac-25.
`https://en.wikipedia.org/wiki/Therac-25`, 2025.
[Online; accessed 27-Jan-2025].

References II

[5] Wikipedia contributors.
Ariane flight V88.
https://en.wikipedia.org/wiki/Ariane_flight_V88, 2025.
[Online; accessed 27-Jan-2025].

[6] Wikipedia contributors.
Heartbleed.
https://en.wikipedia.org/wiki/Heartbleed, 2025.
[Online; accessed 27-Jan-2025].

[7] Wikipedia contributors.
Boeing 737 MAX.
https://en.wikipedia.org/wiki/Boeing_737_MAX, 2025.
[Online; accessed 27-Jan-2025].

References III

[8] Tania Branigan.
Most government IT projects 'not successful'.
https://www.theguardian.com/business/2007/may/18/politics.society, 2007.
[Online; accessed 27-Jan-2025].

[9] Chris Sauer, Andrew Gemino, and Blaize Horner Reich.
The impact of size and volatility on IT project performance.
Communications of the ACM, 50(11):79–84, 2007.

[10] Frederick P Brooks Jr.
The mythical man-month: essays on software engineering.
Pearson Education, 1995.

[11] Frederick Brooks and H Kugler.
No silver bullet.
April, 1987.

References IV

[12] Barry W Boehm.
Software engineering economics.
IEEE transactions on Software Engineering, (1):4–21, 1984.

[13] Georg Kell.
From emissions cheater to climate leader: Vw's journey from dieselgate to embracing e-mobility.
`https://www.forbes.com/sites/georgkell/2022/12/05/`
`from-emissions-cheater-to-climate-leader-vws-journey-from-dieselga`
2022.
[Online; accessed 27-Jan-2025].

[14] Hilary Tuttle.
Facebook scandal raises data privacy concerns.
Risk Management, 65(5):6–9, 2018.

References V

[15] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan.
A survey on bias and fairness in machine learning.
ACM computing surveys (CSUR), 54(6):1–35, 2021.

[16] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner.
Machine bias risk assessments in criminal sentencing.
https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing, 2016.
[Online; accessed 27-Jan-2025].

[17] Adam Rose.
Are face-detection cameras racist?
https://time.com/archive/6906847/are-face-detection-cameras-racist/, 2010.
[Online; accessed 27-Jan-2025].

References VI

[18] Alex De Vries.
Bitcoin's growing energy problem.
Joule, 2(5):801–805, 2018.

[19] Don Gotterbarn, Keith Miller, and Simon Rogerson.
Software engineering code of ethics.
Communications of the ACM, 40(11):110–118, 1997.