Advancing Software Dependability through Understanding Human Errors: A Systematic Framework

Fuqun Huang

Department of Computer Science

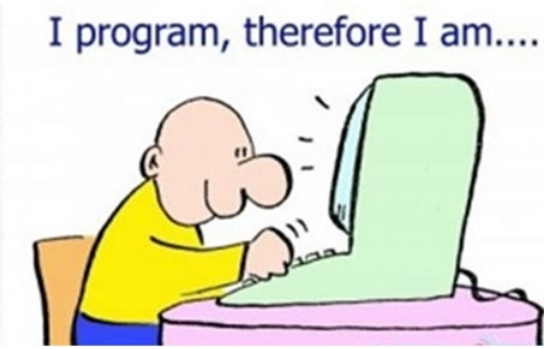
Western Washington University

The 90th Meeting of the IFIP Working Group 10.4

Charlotte, USA, June 25th-28th, 2026

Motivation

- What's software in nature?
- How are software faults caused?



To Err Is Human!

Motivation

- **Design dependable software systems from a fresh perspective:**

- **Programming** is a special writing --Gerald M. Weinberg, "Programming Psychology", 1971

- **A computer program** is by nature a cognitive product that mirrors its designers' thoughts –Fuqun Huang, " Review of Software Psychology", 2013

- **Software defects** are by nature the manifestations of cognitive errors and/or miscommunications between software practitioners--Fuqun Huang, "A Taxonomy System to Identify Human Error Causes for Software Defects", 2012

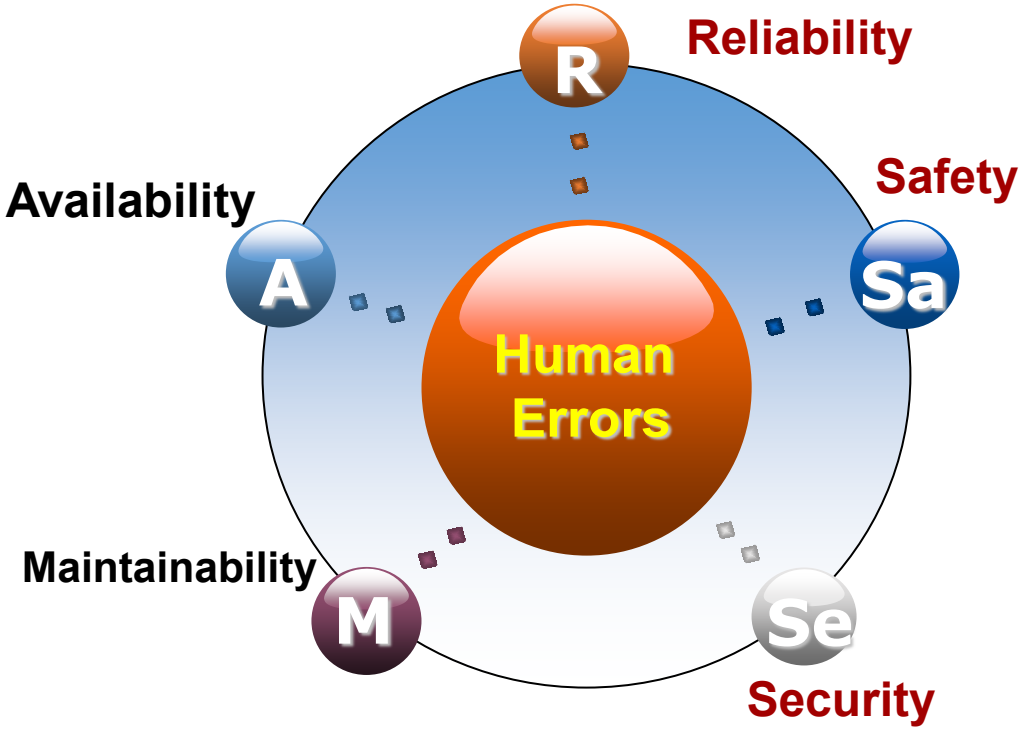
Motivation

- **How to advance software dependability through understanding the human error causes**

Huang, Fuqun, and Bin Liu. "Systematically improving software reliability: Considering human errors of software practitioners." In *23rd Psychology of Programming Interest Group Annual Conference (PPIG 2011)*. 2011.

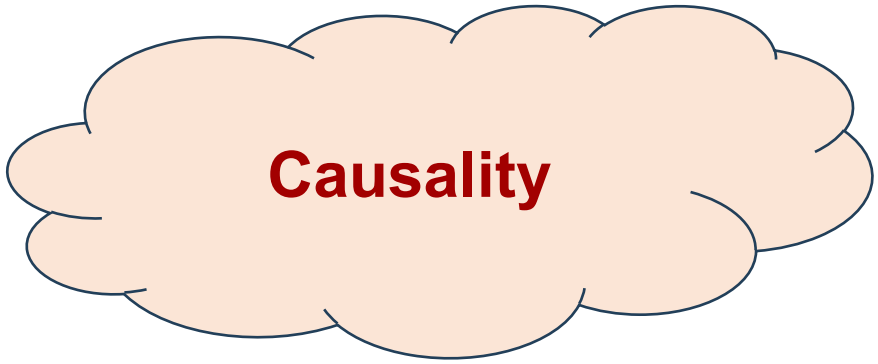
Background

- The Human Errors in Software Engineering Research Group



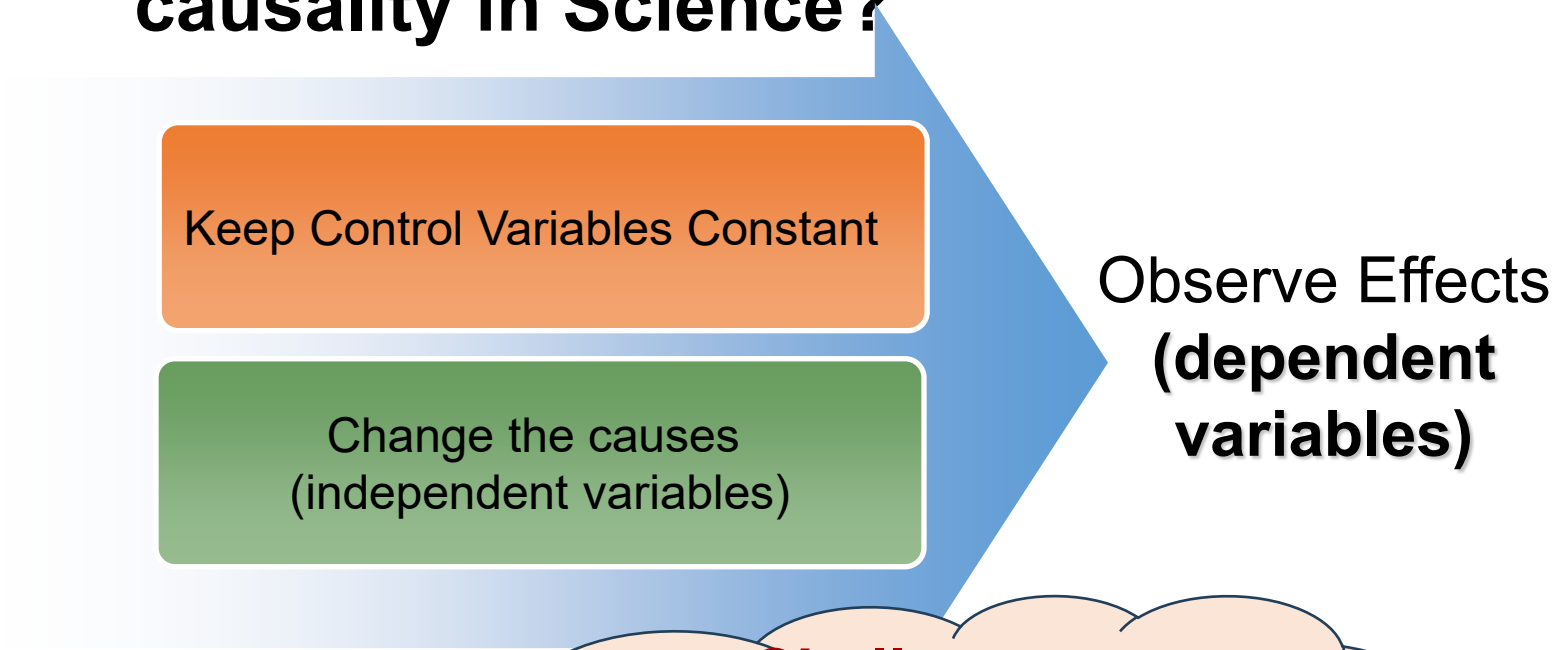
The most challenging questions

Why and how do human errors cause software faults?



Methodology Challenge

- **How are experiments designed for testing causality in Science?**

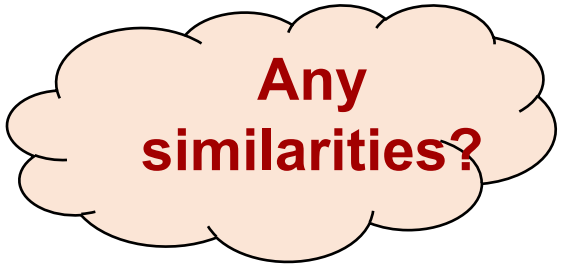


Challenge

Programming largely remains an internal cognitive process

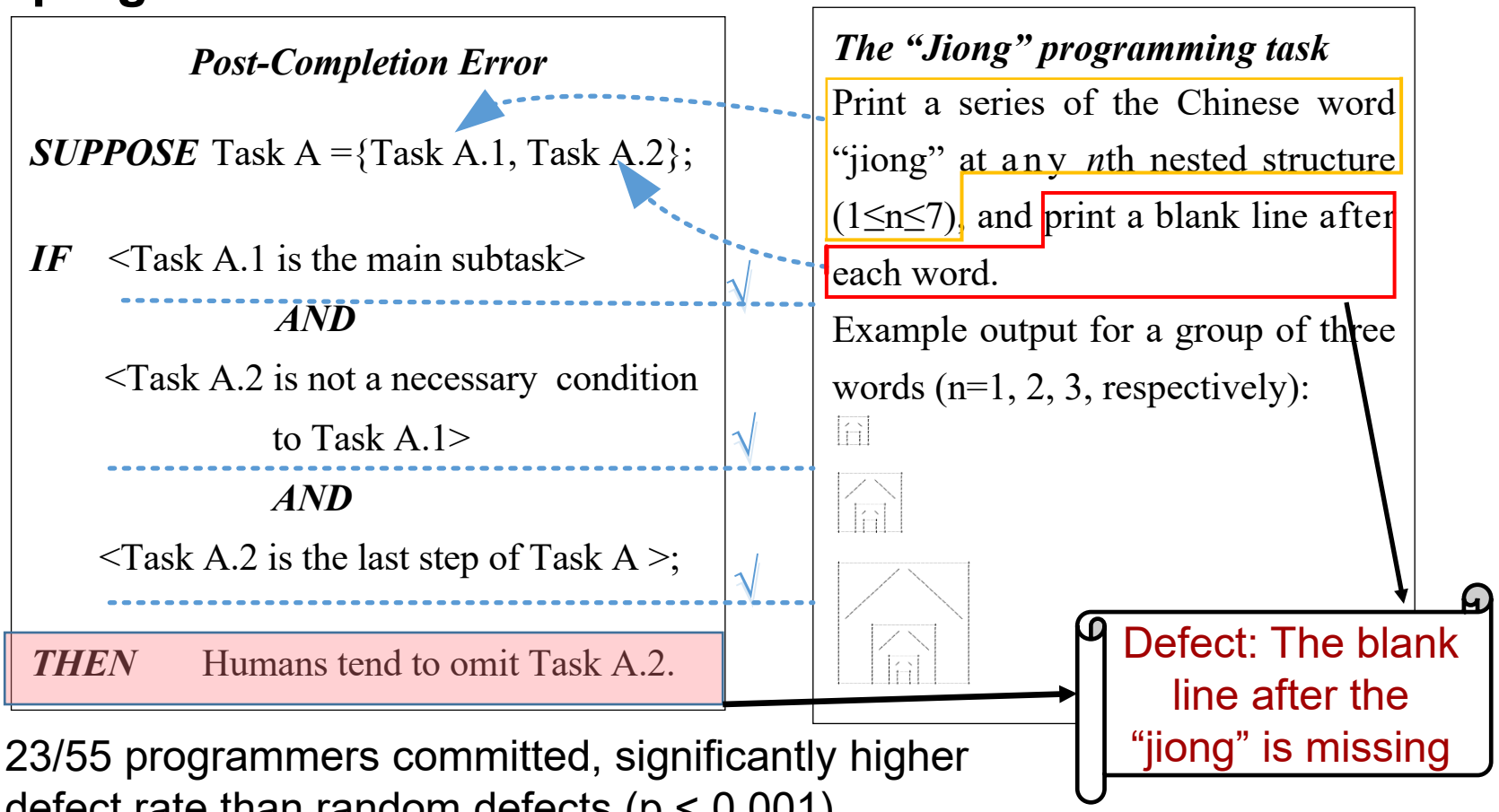
A Peek

- **How many of you have ever**:
  - Forgotten to attach an email attachment?
  - Left the original document in a copy machine?
  - Forgotten to close your garage door?
  - Left a bank card in an ATM after getting cash?



Previous Work

- How Post-Completion Error led to a defect in the Jiong program?



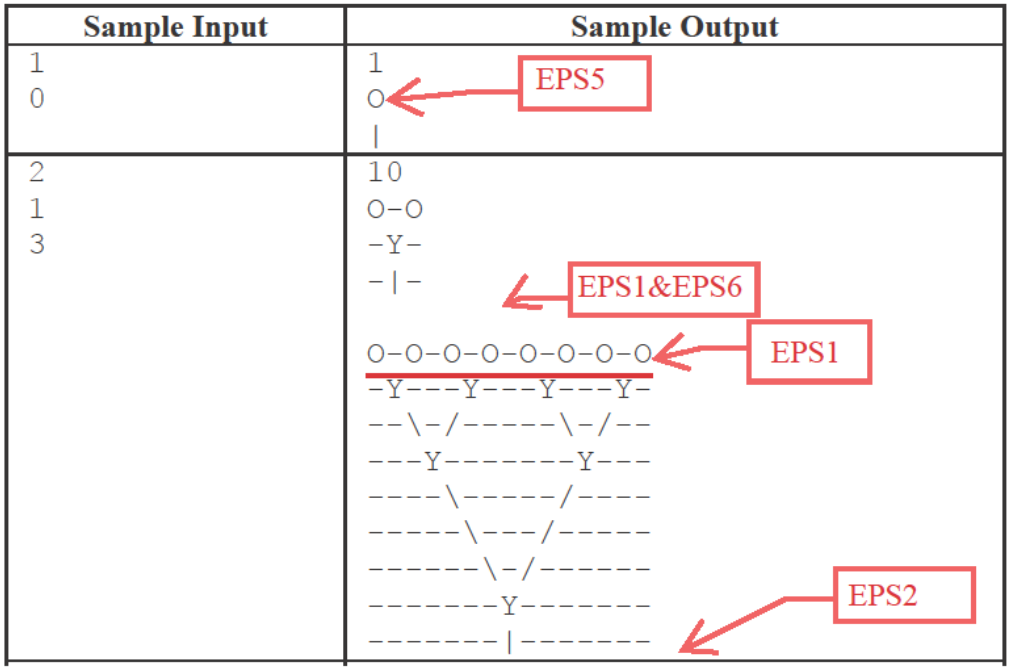
23/55 programmers committed, significantly higher defect rate than random defects ($p < 0.001$)

Causality Experiments

- **"An Experimental Paradigm for Studying Cognitive Mechanisms behind Software Faults", DSN 2026 Disrupt track**

Input: The first line will contain a single integer n: the number of trees in the specified section of the forest. The next line will contain n integers (0 <= n < 20), indicating the age of each tree in years from the starting point. Each age (0<= age <7) will be separated by a single space.

Output: The first line of the output should print the total number of growth buds in the section of the forest **[EPS3].** The remaining output should display a diagram of each tree in the forest, following the specification seen below. Dashes are used below for clarity—each dash should be replaced with an empty space **[EPS4 and EPS6].**



The Magic Trees

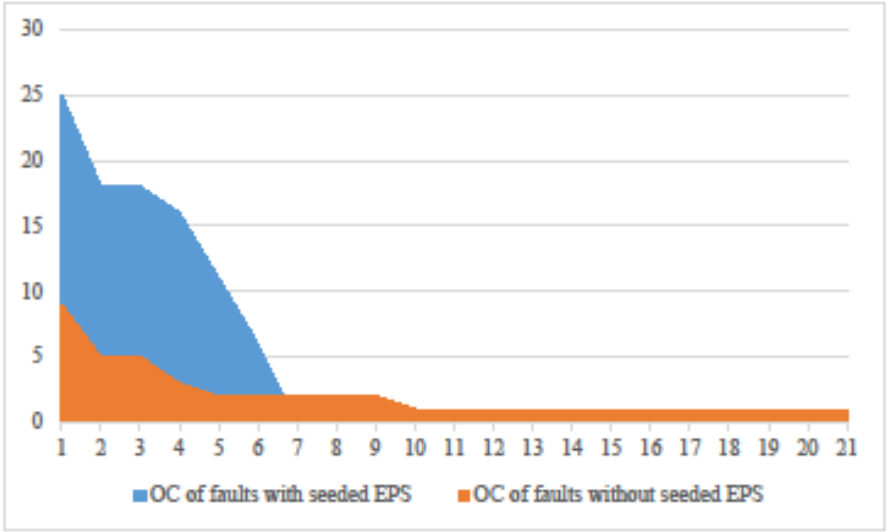
Results

- **Total unique faults:** 27

Publicly available at: https://bit.ly/CoincidentFaultsDataset.

- **Faults at embedded EPS locations**

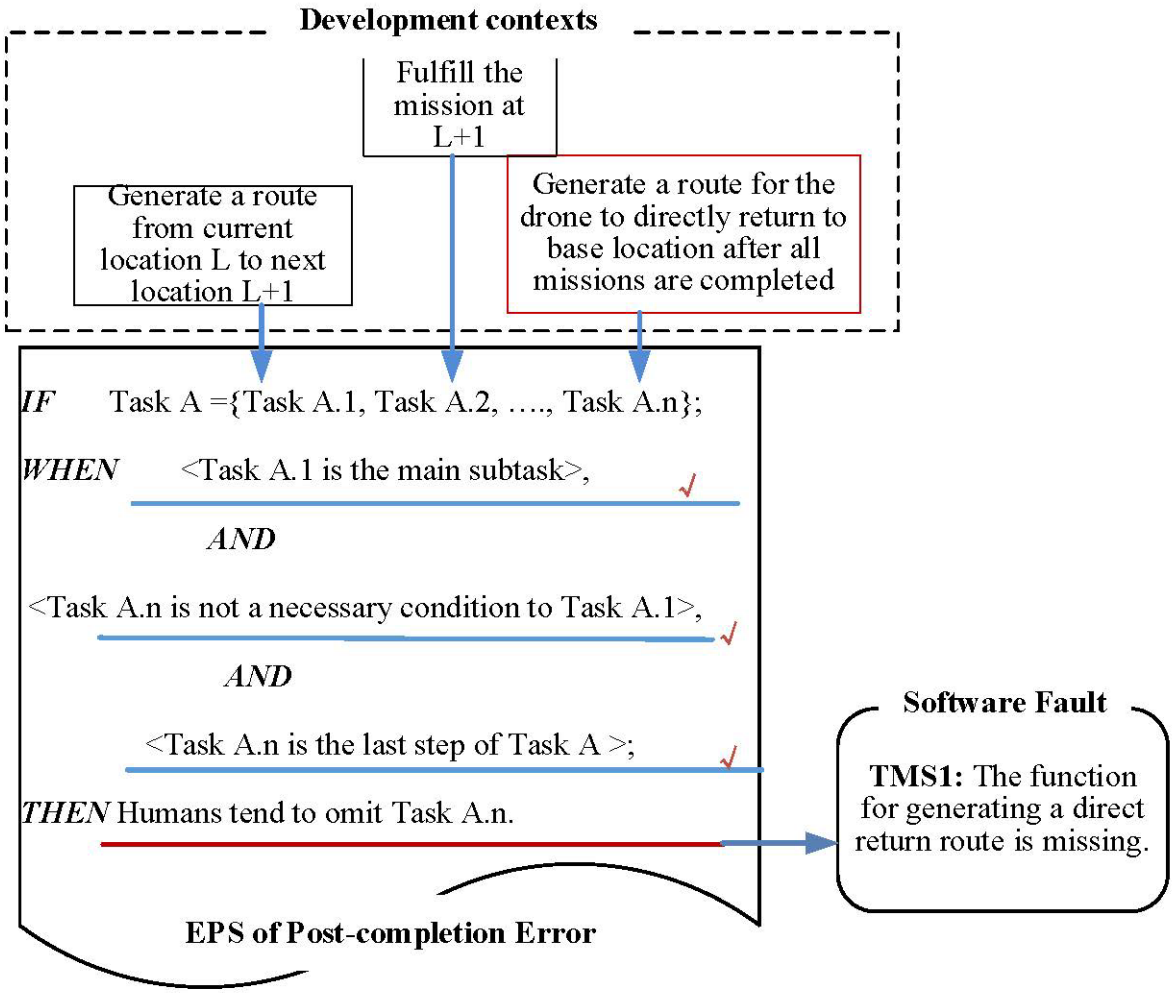
Fault ID	Occurrence	POC
F1	11	25%
F2	18	41%
F3	6	14%
F4	25	57%
F5	16	36%
F6	18	41%



- Prevalence of Occurrences of faults associated with EPSs is significantly higher than that of other faults, **U=125.00, N=27, p < 0.001.**

Empirical Industry Studies

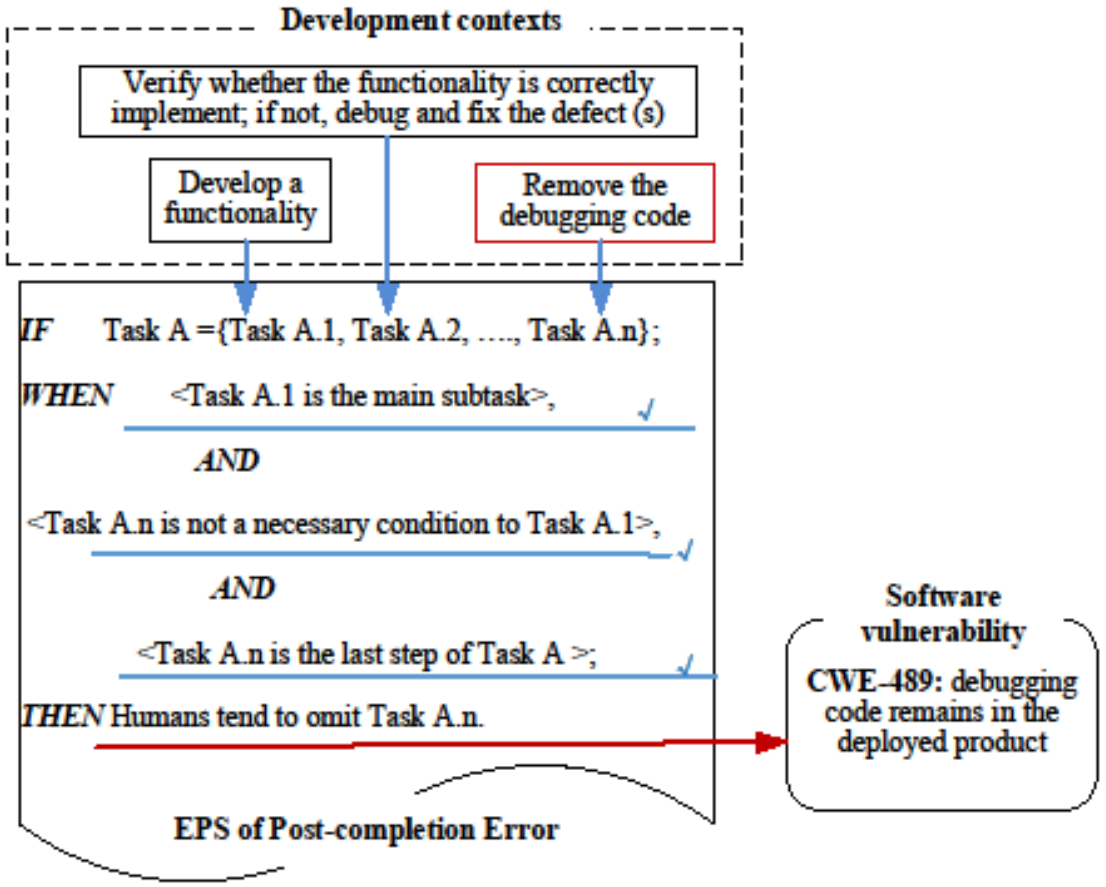
- How PCE manifests as a software fault in a real aircraft system



Huang, Fuqun. "How Post-completion Error Leads to Software Faults and Vulnerabilities: Industrial Case Studies."
In *International Conference on Computer Safety, Reliability, and Security*, pp. 67-81. Cham: Springer Nature Switzerland, 2025.

Empirical Industry Studies

- How PCE manifests as a software vulnerabilities



Huang, Fuqun. "How Post-completion Error Leads to Software Faults and Vulnerabilities: Industrial Case Studies."
In *International Conference on Computer Safety, Reliability, and Security*, pp. 67-81. Cham: Springer Nature Switzerland, 2025.

Empirical Industry Studies

- How a Human Error Mode manifests as multiple software vulnerabilities

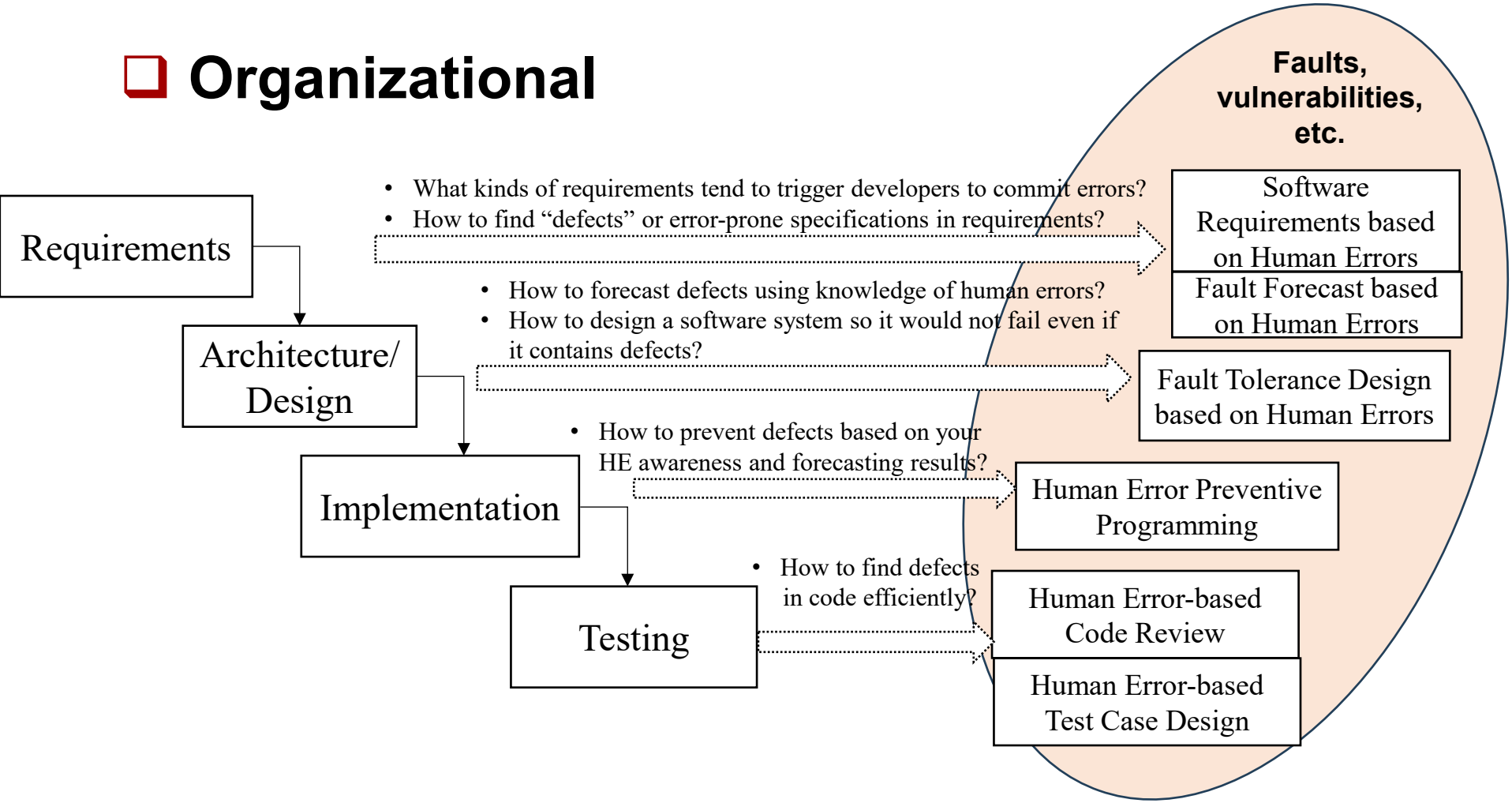
Vulnerability ID	Vulnerability Description	Human Error Root Causes
CWE-212	Forget to remove sensitive information before the product makes the resource available to unauthorized actors.	Post-completion error
CWE-170	The product does not terminate or incorrectly terminates a string or array with a null character or equivalent terminator.	PCE
CWE-460	The product does not clean up its state or incorrectly cleans up its state when an exception is thrown, leading to unexpected state or control flow.	PCE
CWE-489	The product is deployed to unauthorized actors with debugging code still enabled or active.	PCE

The Human-Error-Based Framework for Advancing Software Dependability

(HEFSD)

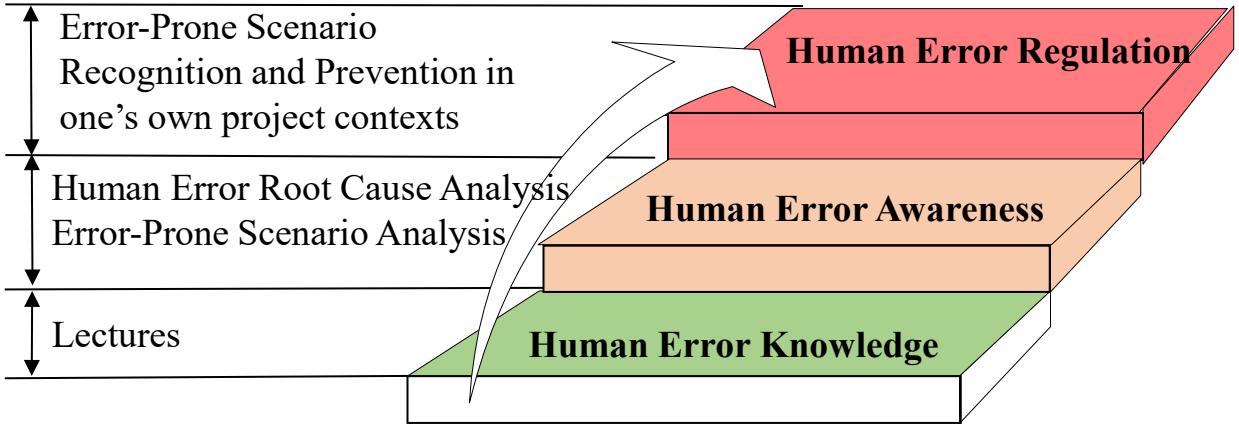
HEFSD

❑ Organizational



The Human Errors in Software Engineering course

❑ Train Individuals



❑ Industrial Consulting
- IBM China Development Center
- Software Certification Center in AVIC

❑ Permanent stack course, Western Washington University, 2024-current

❑ Independent Study, University of Coimbra, 2022

- The Human Errors in Software Engineering course, The 38th International Conference on Software Engineering Education and Training (CSEET) 2026.
- A New Graduate Course Structure for Addressing Human Errors in Software Development, IEEE FIE 2024

Look Forward

- **Invited Talk** "Allowing Students to Use ChatGPT? Whether, When, and How through the Lens of Programming Cognition" in the 38TH Conference on Software Engineering Education and Training (CSEE&T), Florence, July 20-22.
- **Tutorial "**Human Error Prevention for Trustworthy Cyber-Physical Systems", IEEE International Conference on Systems, Man, and Cybernetics (SMC 2026), Oct. 4-7, Bellevue, WA.
- **Book** "Human Errors in Software Engineering", in publication with Springer

Questions?

Fuqun Huang, PhD

Department of Computer Science

Western Washington University

huangf2@wwu.edu