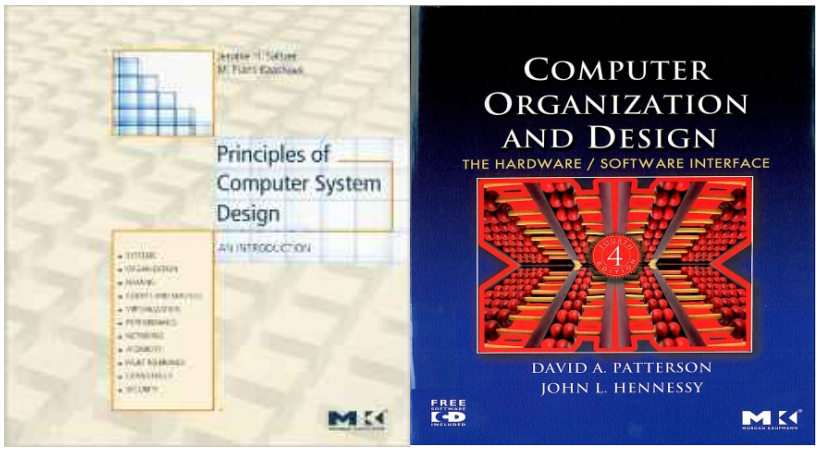
# Computer System Engineering

## Lecture 4: Fault - Tolerance




**Nguyen Minh Son, Ph.D**

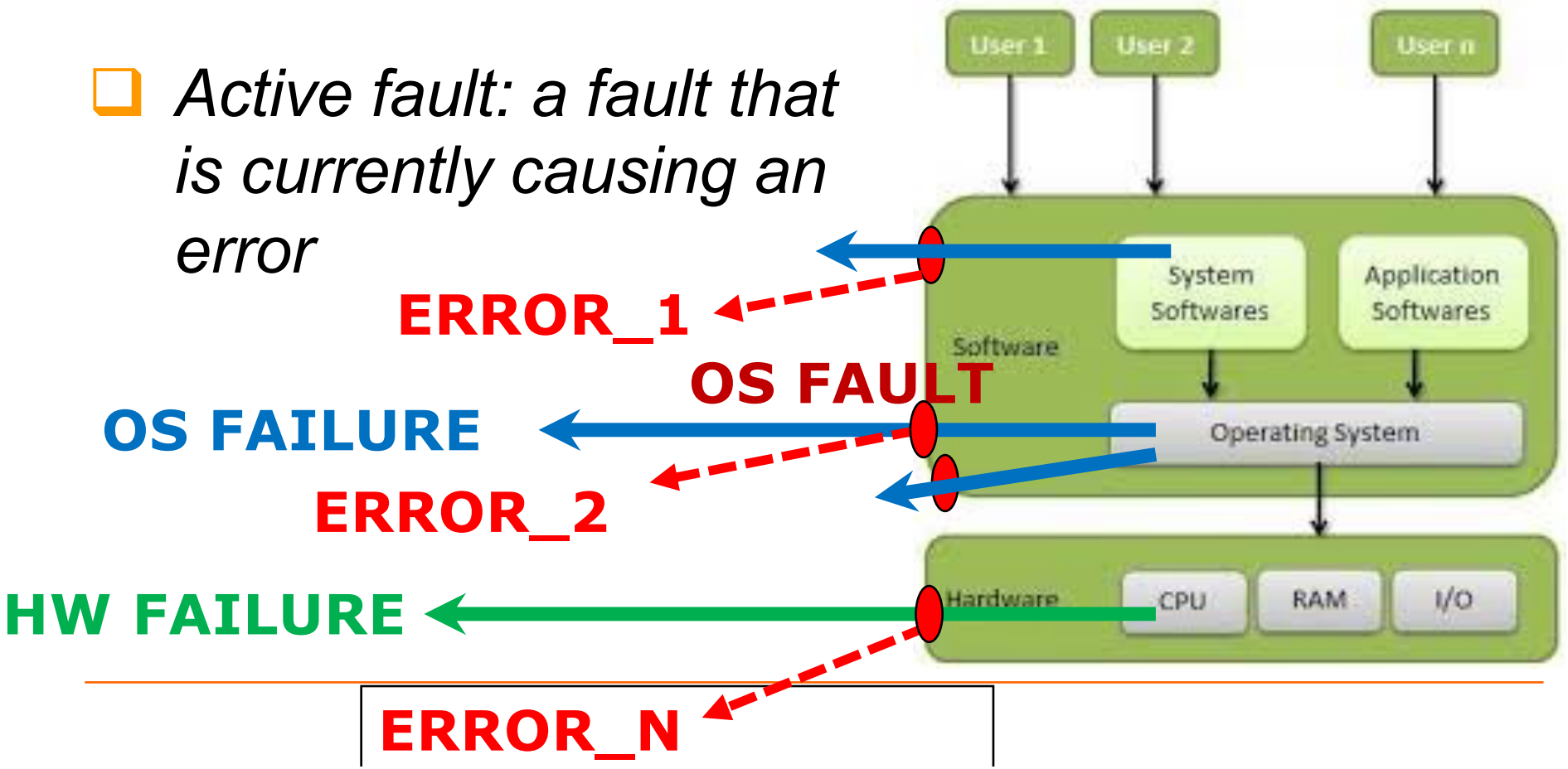
UIT-HCM

# Outline

- Faults, failures, and fault-tolerant design
- Measures of reliability and failure tolerance
- Tolerating active faults
- Systematically applying redundancy
- Applying redundancy to software and data
- Conclusions

# Faults, failures, and fault-tolerant design

- *Fault: a defect in materials, design, or implementation that may (or may not) cause an error* and lead to a failure
- *Active fault: a fault that is currently causing an error*



# Faults, failures, and fault-tolerant design

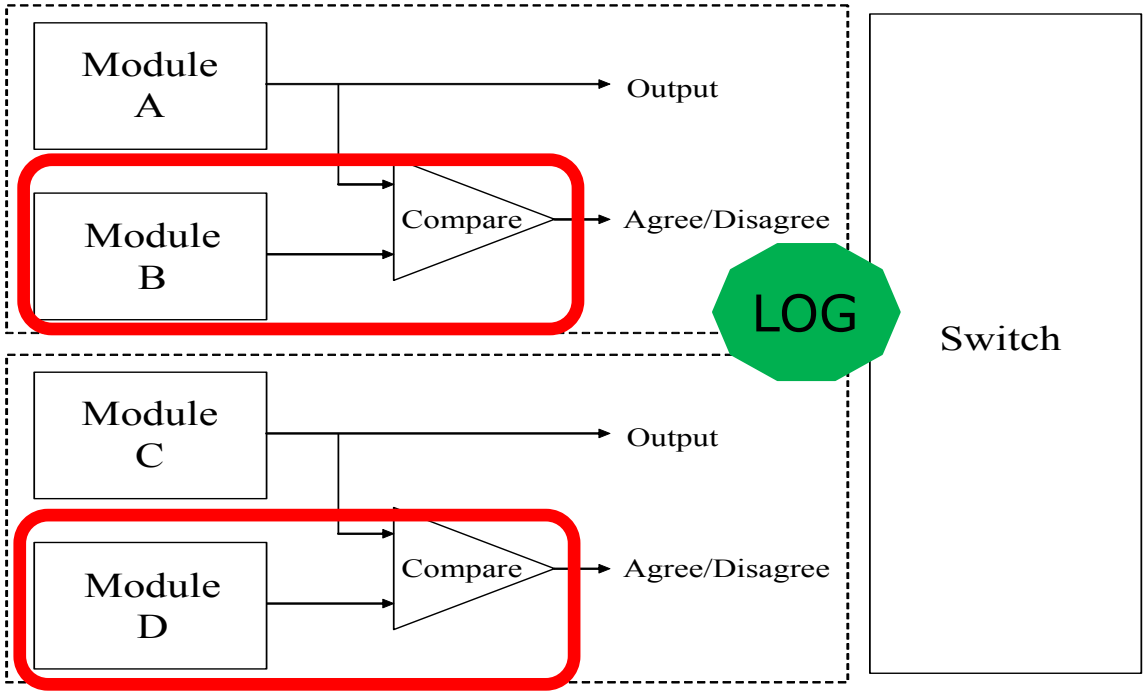
- *Error:* Informally, a label for an incorrect data value or control signal caused by an active fault. If there is a complete formal specification for the internal design of a module, an error is a violation of some assertion or invariant of the specification.

- *Failure:* The outcome when a component or system does not produce the intended result at its interface.

# Faults, failures, and fault-tolerant design

- *Fault avoidance:* all components are reliable

- *Fault tolerance:* collection of techniques to build reliable systems from unreliable components



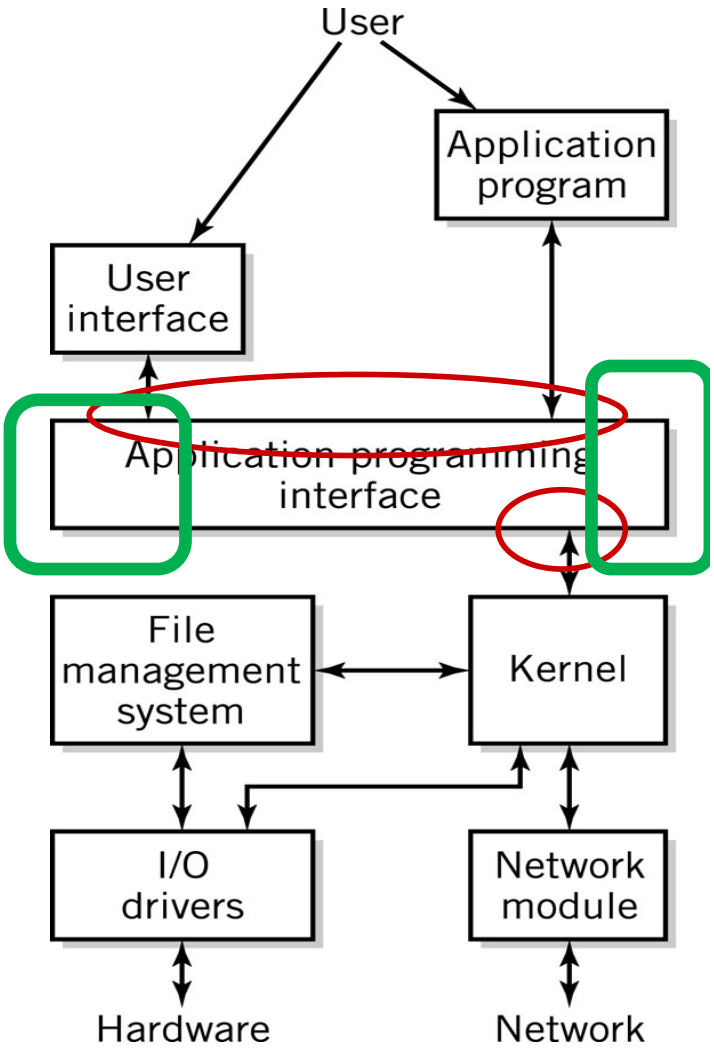
# The *fault-tolerance design process*

1. Develop a fault-tolerance model

2. Apply modularity to contain the damage from the high-risk errors

3. Design and implement procedures that can mask the detected errors (temporal/spatial redundancy)

4. Update the fault-tolerance model to account for those improvements

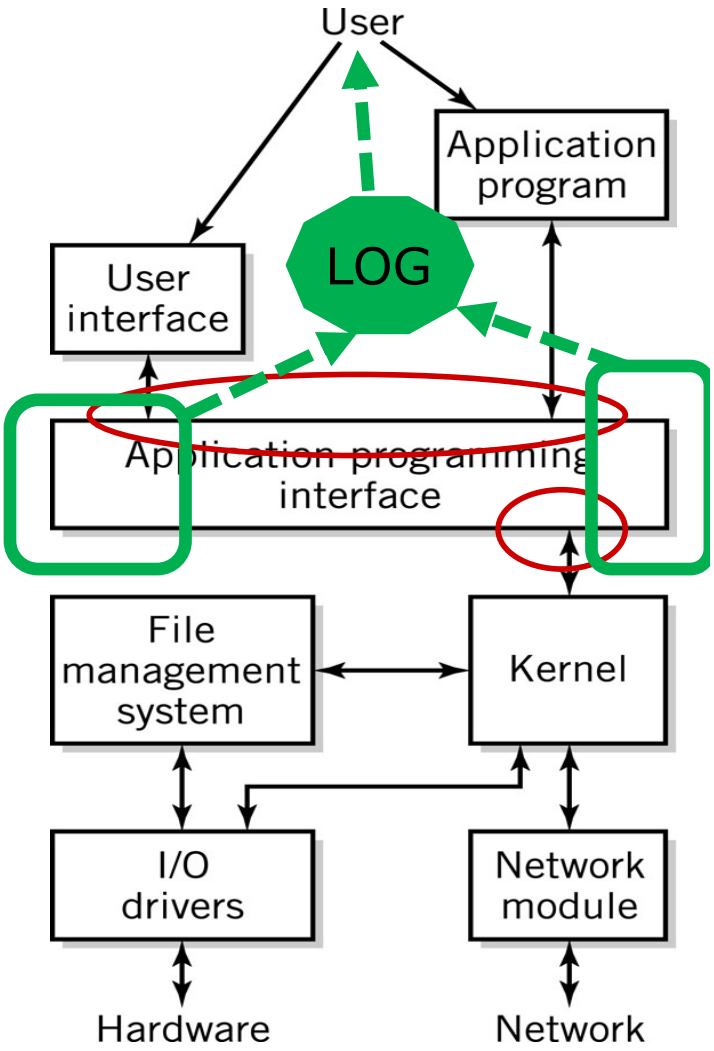


# The *fault-tolerance design process*

5. Iterate the design and the model until the probability of untolerated faults is low enough

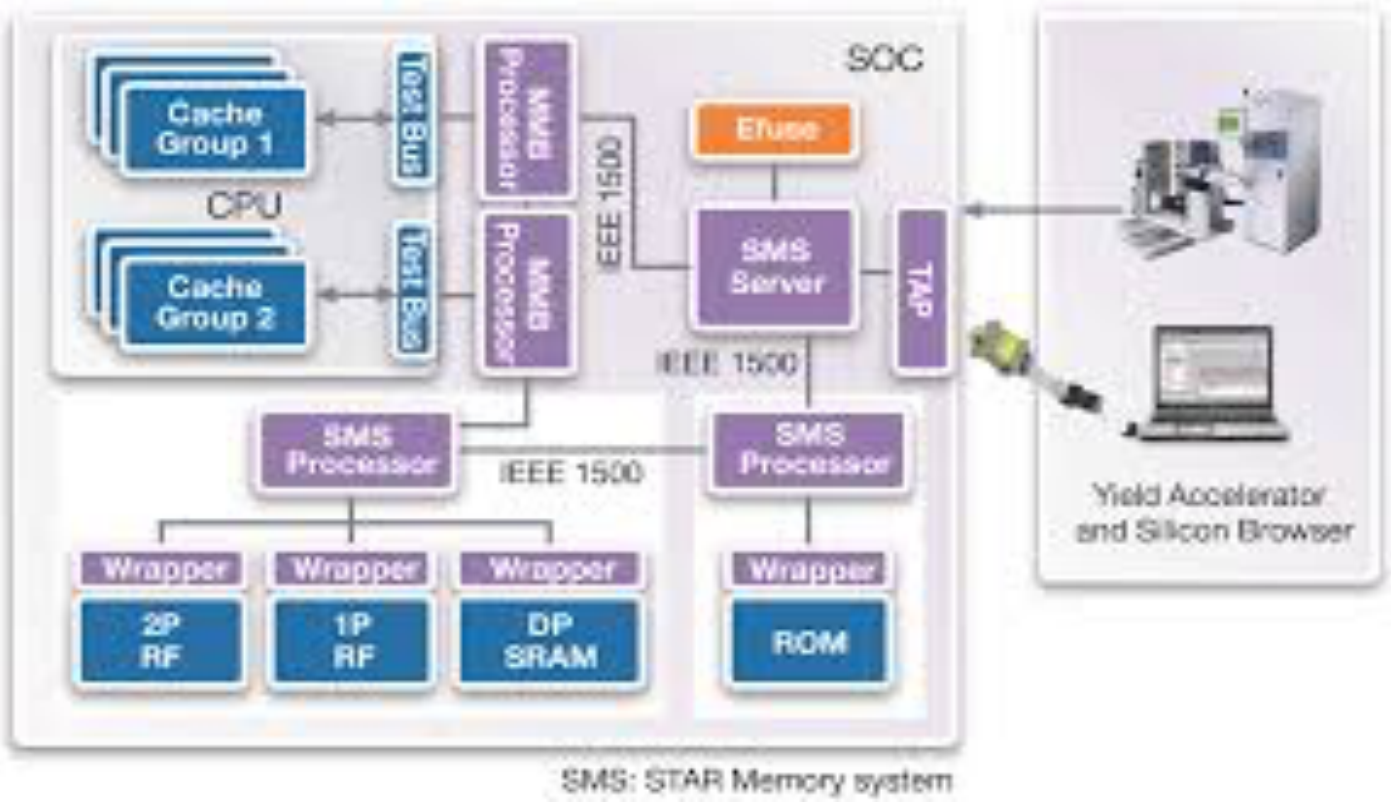
6. Observe the system in the field

7. Use the logs of masked faults and the postmortem reports about failures to revise and improve the fault-tolerance model and reiterate the design



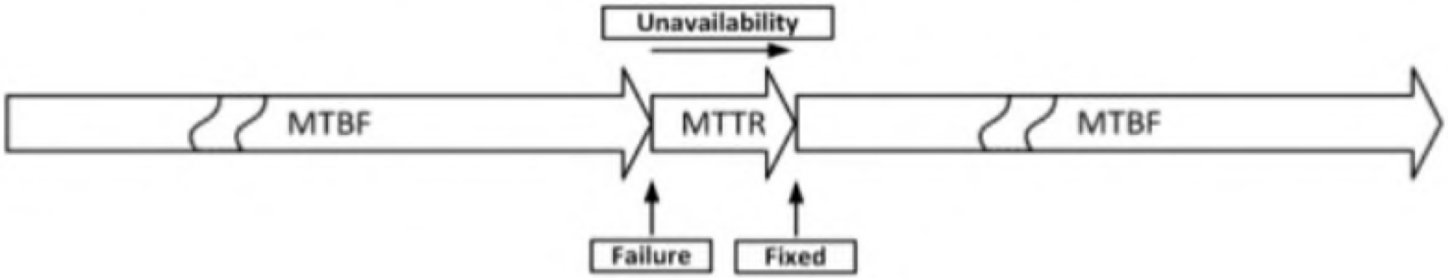
# Faults, failures, and fault-tolerant design

- Software fault
- Hardware fault
- Design fault
- Implementation fault
- Operations fault
- Environment fault



# Measures of reliability and failure tolerance

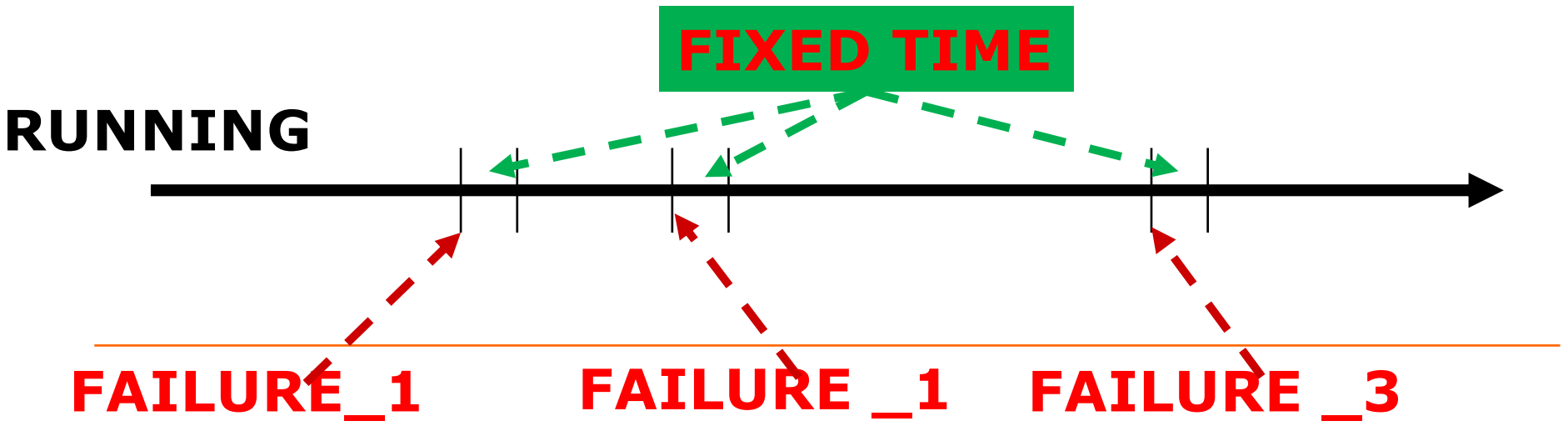
- *Availability*: A measure of the time that a system was actually usable, as a fraction of the time that it was intended to be usable.
  - The time to failure (TTF)
  - The time to repair (TTR)
  - The mean time to failure (MTTF)
  - The mean time to repair (MTTR)
  - The mean time between failures (MTBF)



# Measures of reliability and failure tolerance

$$Availability = \frac{\text{time system was running}}{\text{time system should have been running}} = \frac{\sum_{i=1}^{N} TTF_i}{\sum_{i=1}^{N} (TTF_i + TTR_i)}$$

$$Availability = \frac{MTTF}{MTBF} = \frac{MTTF}{MTTF + MTTR} = \frac{MTBF - MTTR}{MTBF}$$



# Measures of reliability and failure tolerance

$$Availability = \frac{\text{time system was running}}{\text{time system should have been running}} = \frac{\sum_{i=1}^{N} TTF_i}{\sum_{i=1}^{N} (TTF_i + TTR_i)}$$

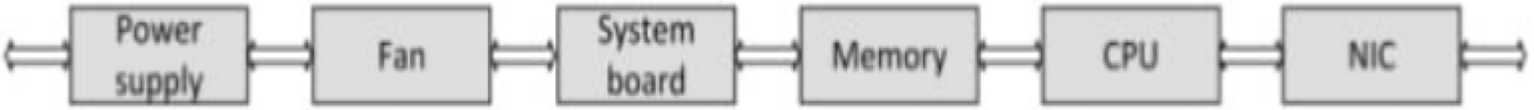
| Availability % | Downtime per year | Downtime per month | Downtime per week |
|---|---|---|---|
| 99.8% | 17.5 hours | 86.2 minutes | 20.2 minutes |
| 99.9% ("three nines") | 8.8 hours | 43.2 minutes | 10.1 minutes |
| 99.99% ("four nines") | 52.6 minutes | 4.3 minutes | 1.0 minutes |
| 99.999% ("five nines") | 5.3 minutes | 25.9 seconds | 6.1 seconds |

$$Availability = \frac{MTTF}{MTBF} = \frac{MTTF}{MTTF + MTTR} = \frac{MTBF - MTTR}{MTBF}$$

| Component | MTBF (hours) |
|---|---|
| Hard disk | 750,000 |
| Power supply | 100,000 |
| Fan | 100,000 |
| Ethernet Network Switch | 350,000 |
| RAM | 1,000,000 |

# Measures of reliability and failure tolerance

$$Availability = \frac{MTTF}{MTBF} = \frac{MTTF}{MTTF + MTTR} = \frac{MTBF - MTTR}{MTBF}$$




| Component | MTBF (h) | MTTR (h) | Availability | in % |
|---|---|---|---|---|
| Power supply | 100,000 | 8 | 0.9999200 | 99.99200 |
| Fan | 100,000 | 8 | 0.9999200 | 99.99200 |
| System board | 300,000 | 8 | 0.9999733 | 99.99733 |
| Memory | 1,000,000 | 8 | 0,9999920 | 99.99920 |
| CPU | 500,000 | 8 | 0.9999840 | 99.99840 |
| Network Interface Controller (NIC) | 250,000 | 8 | 0.9999680 | 99.99680 |

Example: the above system's availability is:

$$0.9999200 \times 0.9999200 \times 0.9999733 \times 0.9999920 \times 0.9999840 \times 0.9999680 = 0.99977 = \mathbf{99.977\%}$$

# Conclusions

- Design principles
  - be explicit
  - design for iteration
  - keep digging
  - the safety margin
  - adopt sweeping simplifications
  - Deterioration and corruption accumulate unnoticed—until the next use

# Enjoy !!!

Q&A

# Exercise

| | Hệ thống máy tính 1 | | Hệ thống máy tính 2 | | Hệ thống máy tính 3 | |
|---|---|---|---|---|---|---|
| | Thực thi | Khởi động | Thực thi | Khởi động | Thực thi | Khởi động |
| **Chương trình A** | 1500s | 50s | 5000s | 5s | 2000s | 10s |
| **Chương trình B** | 2500s | 50s | 4000s | 10s | 5000s | 20s |
| **Chương trình C** | 15000s | 50s | 18500s | 10s | 9000s | 50s |

Trong 3 máy tính trên, máy tính nào có hiệu suất tối ưu nhất? Tại sao?