CSED601
Dependable Computing
Lecture 2

Jong Kim

Dept. of CSE

POSTECH

What is the difference?

- Failure (External universe)
  - – The delivered service deviates from the specified service.
- Error (information universe)
  - – An error is that part of the system state which is liable to lead to failure.
- Fault (Physical universe)
  - – The cause of a fault

Techniques for dependable computing

- Fault-avoidance
  - Prevent the occurrence of a fault
- Fault-tolerance
  - Providing a specified service in spite of faults occurrence.
- Error-removal
  - Minimize the presence of latent fault.
- Error-forecasting
  - Estimating the presence, the creation, and the consequences of errors

Cause of faults

- Specification mistakes
  - Specification/Design errors
- Implementation mistakes
  - Misuse of tools
- Component defects
  - Physical worn out, aging
- External disturbance
  - External noise, E-M signal

Sources of Downtime

Category	Early 80's	late 80's	90's
Hardware + Environment	32%	29%	20%
Software	26%	58%	40%
Human Operators	42%	13%	40%

* Data from Stanford lecture

- Is software getting worse?

Is Software Getting Worse?

- Known data
  - Tandem OS (1985): 4 MLOC
  - Linux (2001): 30MLOC
  - Windows XP (2001): 40-50 MLOC
- Gray’s estimate: 1 bug/KLOC
- Reducing bugs/KLOC vs. increasing KLOCs/product

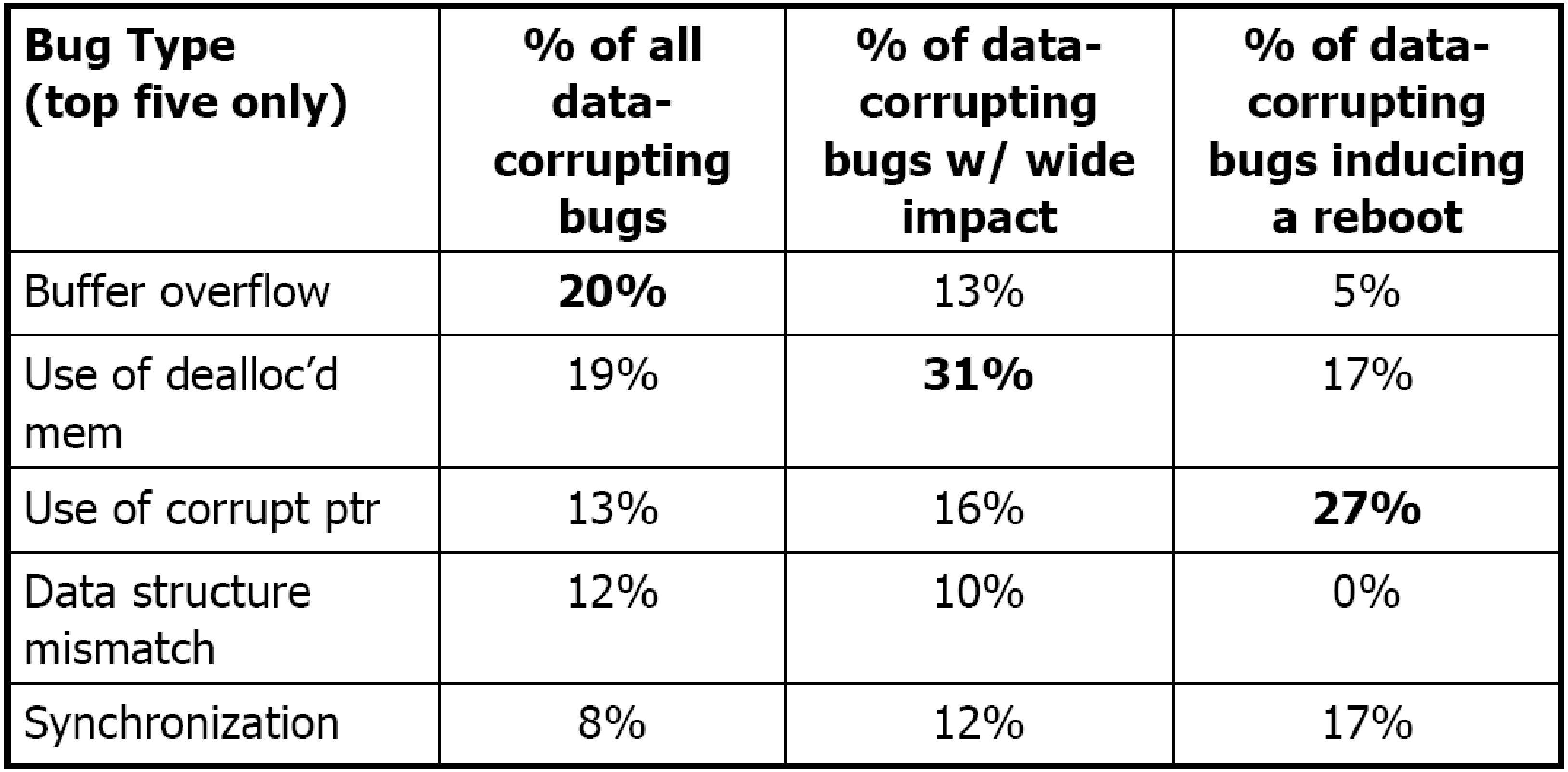
Classification of faults

- By fault nature:
  - H/W (analog or digital) or S/W
- By fault duration:
  - Permanent or Intermittent or Transient
- By fault value:
  - Determinate or Indeterminate
- By fault extent:
  - Local or Global

Software Failures

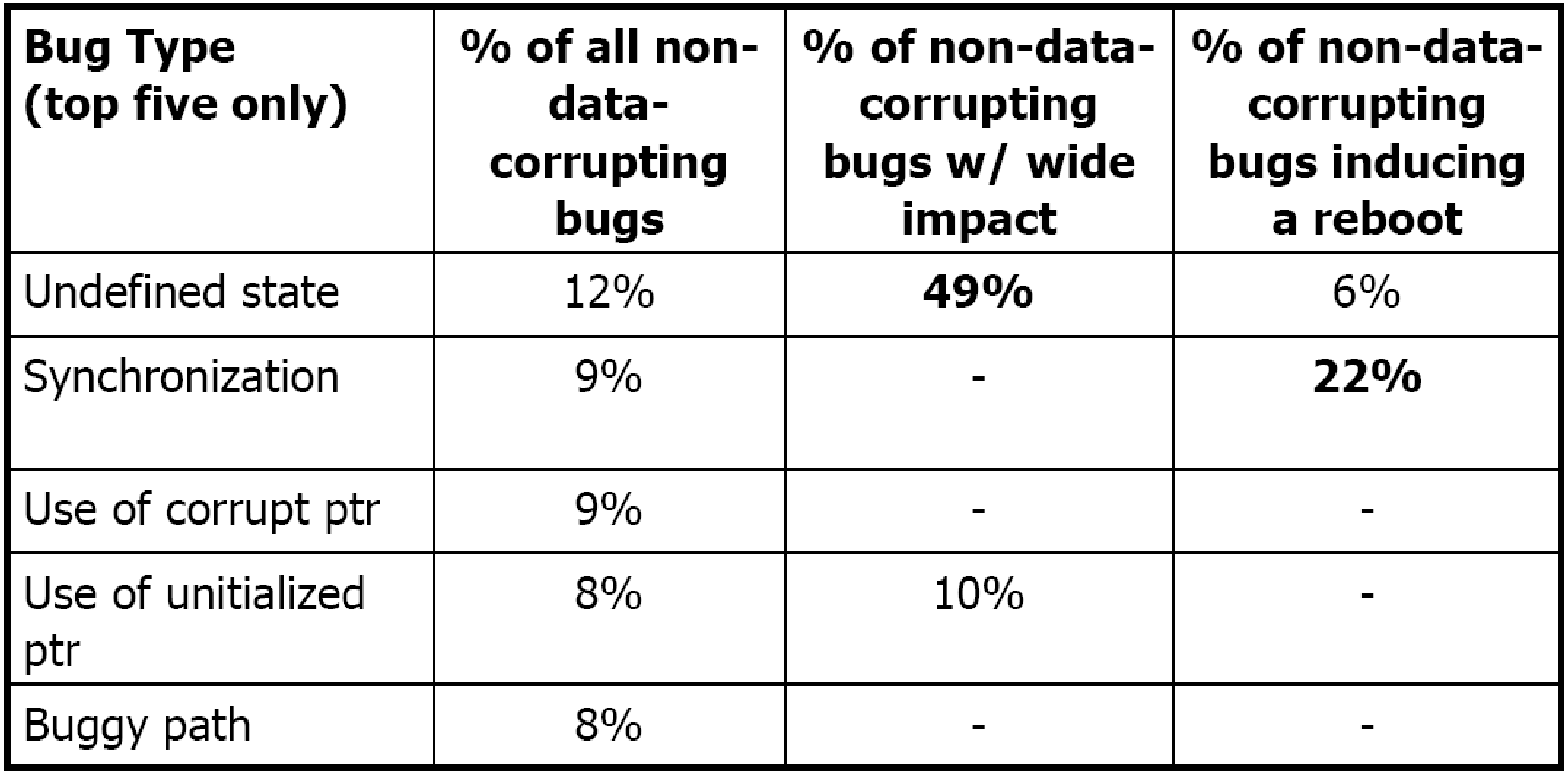
- Crash
- Hang
- Respond correctly but too late
- Provide wrong data

Data Corrupting Bugs



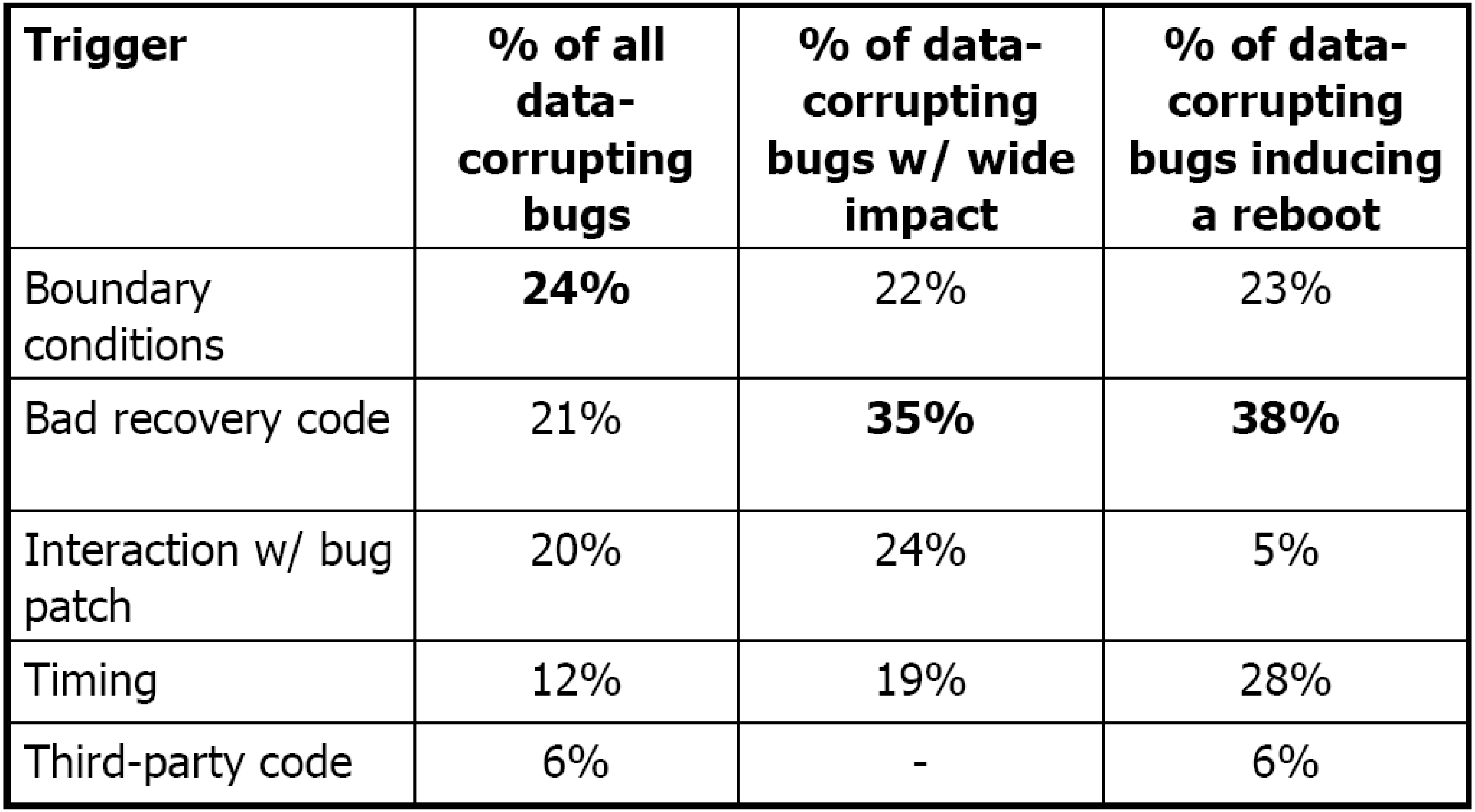
Bug Type (top five only)	% of all data-corrupting bugs	% of data-corrupting bugs w/ wide impact	% of data-corrupting bugs inducing a reboot
Buffer overflow	**20%**	13%	5%
Use of dealloc'd mem	19%	**31%**	17%
Use of corrupt ptr	13%	16%	**27%**
Data structure mismatch	12%	10%	0%
Synchronization	8%	12%	17%

Non-Data Corrupting Bugs



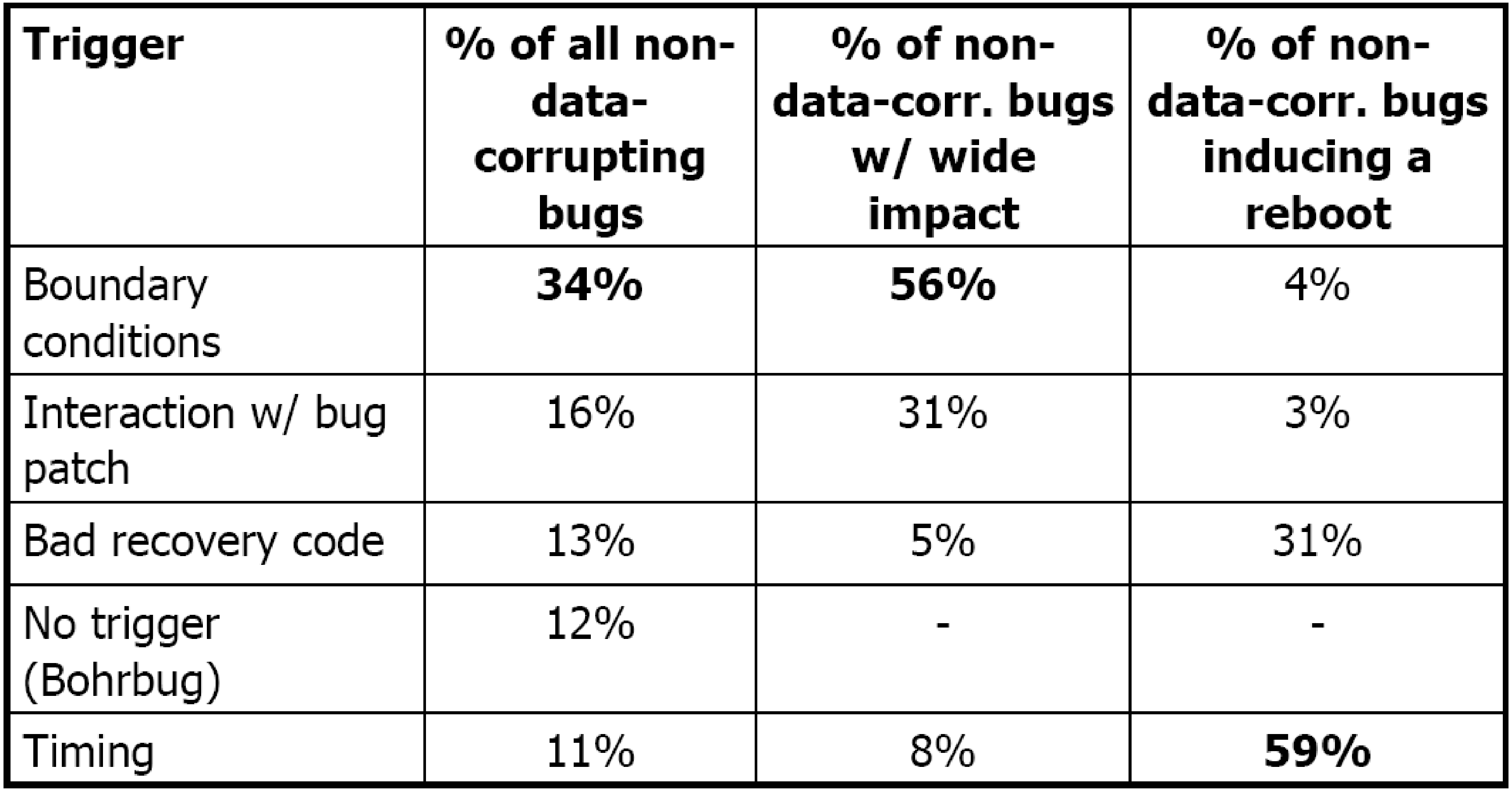
Bug Type (top five only)	**% of all non-data-corrupting bugs**	**% of non-data-corrupting bugs w/ wide impact**	**% of non-data-corrupting bugs inducing a reboot**
Undefined state	12%	**49%**	6%
Synchronization	9%	-	**22%**
Use of corrupt ptr	9%	-	-
Use of unitialized ptr	8%	10%	-
Buggy path	8%	-	-

Top-5 Triggers: Data Corrupting Bugs



Trigger	**% of all data-corrupting bugs**	**% of data-corrupting bugs w/ wide impact**	**% of data-corrupting bugs inducing a reboot**
Boundary conditions	**24%**	22%	23%
Bad recovery code	21%	**35%**	**38%**
Interaction w/ bug patch	20%	24%	5%
Timing	12%	19%	28%
Third-party code	6%	-	6%

Top-5 Triggers: Non-Data Corrupting Bugs



Trigger	**% of all non-data-corrupting bugs**	**% of non-data-corr. bugs w/ wide impact**	**% of non-data-corr. bugs inducing a reboot**
Boundary conditions	**34%**	**56%**	4%
Interaction w/ bug patch	16%	31%	3%
Bad recovery code	13%	5%	31%
No trigger (Bohrbug)	12%	-	-
Timing	11%	8%	**59%**

Fault Models

- Needed
  - to define the types of faults that will be considered
  - to define the behavior these faults will have
  - to make problems tractable
- Modeling level
  - Circuit-level : short or open
  - Logical-level : stuck-at-0 or stuck-at-1
  - Register-Transfer level : working/failed
  - System-level : design / naturally / artificial

Digital system Fault Modeling

- Manufacturing stages
  - – Design maturity testing
  - – Incoming inspection
  - – Process maturity testing (burn-in test)
- Operational life stages
  - – Infant mortality period
  - – Steady-state stress
  - – Wear-out period

Fault Avoidance

- Informal definition
  - Any techniques that is used to prevent faults in the first place.
- Methods
  - Design review
  - Component testing
  - Other quality control methods
  - Shielding

Errors

- When does it happens
  - When faults in a system affect the information in the way that the information differs with the specified behavior. (information universe)
- Sources of errors
  - Permanent fault : low
  - Intermittent fault : high
  - Transient fault : low

Error Model

- Error characteristic
  - Information change in undesired way
- Error model
  - Change of truth table (logical value change)

Fault Masking

- Informal definition
  - Any process that prevents faults in a system from introducing errors into the information structure of that system.
- Methods
  - Error-correcting memory
  - Majority voting

Fault Tolerance

- Informal definition
  - The ability of a system to continue to perform its tasks after the occurrence of faults.
  - Consists of a series of actions
    - Fault-detection
    - Fault-location
    - Fault containment
    - Damage assessment
    - Reconfiguration
    - Recovery

Homework #1

- Submit a summary of the paper (max. 4 pages)
- Investigate/find a fault/error model on Blockchain or IoT that is not common in general systems
- Investigate what dependability aspect(s) is(are) important in blockchain or IoT
- Due on Sep. 12 (one week)