CSED601
Dependable Computing
Lecture 1

Jong Kim

Dept. of CSE

POSTECH

References

- Anh Nguyen-Tuong, University of Virginia http://www.cs.virginia.edu/~zw4j/cs656/DependableComputing.ppt

- Algirdas Avizienis, Jean-Claude Laprie, Brian Randell, and Carl Landwehr, Basic Concepts and Taxonomy of Dependable and Secure Computing, IEEE Transactions on Dependable and Secure Computing, Vol. 1, No. 1, Jan-Mar 2004.

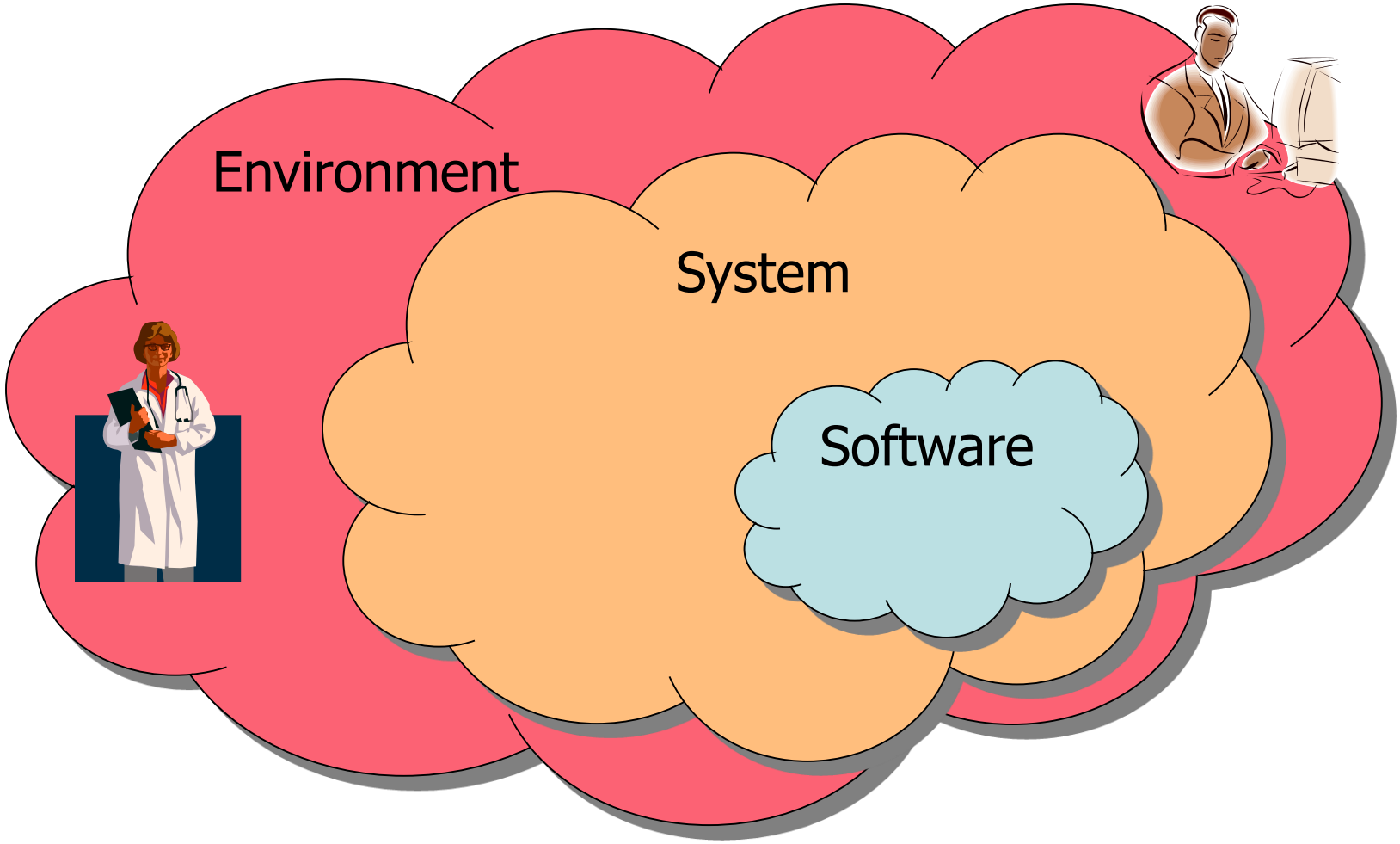
Goals

- Understanding basic concepts and terminology associated with dependable computing

- View current events and OS concepts with “dependability optic”

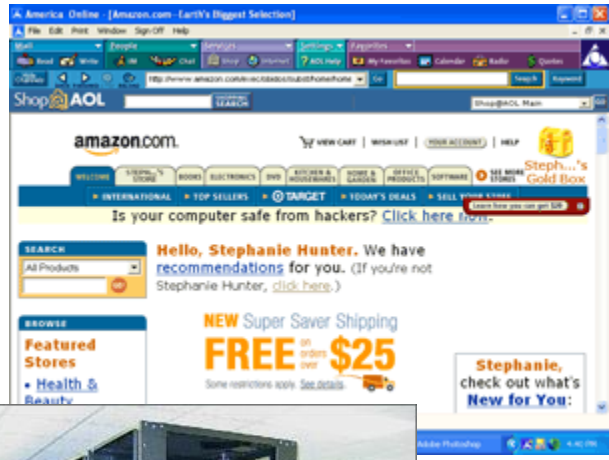

Rapid
A fatal exception 0E has occurred at 0028:C0011E36 in VXD VMM(01) +
00010E36. The current application will be terminated.
* Press any key to terminate the current application.
* Press CTRL+ALT+DEL again to restart your computer. You will
lose any unsaved information in all applications.
Press any key to continue _

Systems and Software



Titan IVB Launch - USAF Photo

TiVo Series2 DVR
140 Hour capacity drive



Designed exclusively
for driving pleasure:
the BMW Z4.

Consequences of Failure

- Injury or loss of life
- Environmental damage
- Damage to or loss of equipment
- Financial loss:
  - Theft
  - Useless or defective mass-produced equipment
  - Loss of production capacity or service
  - Loss of business reputation, customer base

Consequences of Failure

- Loss of revenue
  - Software bugs: $200 billion/year (SCC)
    - Note: recent worms and viruses
  - 1 hour of downtime costs (InternetWeek, 2000):
    Brokerage operations $6,450,000/hr
    Credit card auth. $2,600,000/hr
    Ebay (1 outage/22hrs) $225,000/hr
    Home Shopping Channel $113,000/hr
    Airline reservation ctr $89,000/hr
    ATM services fee $14,000/hr
  - Note: Ebay (22 hrs, 1999)
    - $4 Billion Market Cap loss

Facets of dependability

- De jure and de facto taxonomy (Laprie)
  - Reliability → continuity of correct service
  - Availability → readiness for usage
  - Safety → no catastropic consequences
  - Security → prevention unauthorized access
    - Integrity, Confidentiality
  - Maintainability → repair and modification

Customers *must* identify the dependability requirements of their system and developers must design so as to achieve them

Historical Evolution of Concerns

- 40's: ENIAC
  - 18K vacuum tubes → failed ~ every 7 mns
  - 18K multiplies/minute → 7 mns ~ one program execution

  Need RELIABILITY
- 60's: Interactive systems
  + AVAILABILITY
- 70's: F-8 Crusader, MD-11, Patriot missile defense
  + SAFETY
- 90's-today: Internet, E-commerce, Grid/Web services
  + SECURITY

Terminology

- We need to be able to communicate in a precise manner:
  - Researchers
  - Developers
  - Customers
- There are everyday notions of these terms
- The public has an interest
- But public terminology is imprecise

Reliability

Rel(t) = Probability that the system will operate correctly in a specified operating environment up until time t

Mean Time To Failure

MTTF = Expected Value[Rel(t)]

- Note that t is important
- If a system only needs to operate for ten hours at a time, then that is the reliability target

Recoverability

Rec(t) = Probability that the system will operate correctly at time t after failure

Mean Time To Repair:

MTTR = Expected Value[Rec(t)]

Availability

A(t) = Probability that the system will be operational at time t

$$E[A(t)] = MTTF / (MTTF + MTTR)$$

- Literally, readiness for service
  - Only applies when you ask for a service
- Admits the possibility of brief outages
- Fundamentally different concept than Reliability

Nines of availability

# 9's	%	Downtime / year	Systems
2	99%	~5000 mns	General web site
3	99.9%	~500 mns	Amazon.com
4	99.99%	~50 mns	Enterprise server
5	99.999%	~5 mns	Telephone System
6	99.9999%	~30 sec	Phone switches

Caveats: How measured? What does it mean to be operational?

Reliability vs. Availability

- They are not the same.....
- Example:

  A system that fails, on average, once per hour but which restarts automatically in ten milliseconds is not very reliable but is highly available

  Availability = 0.9999972

Design Tradeoffs

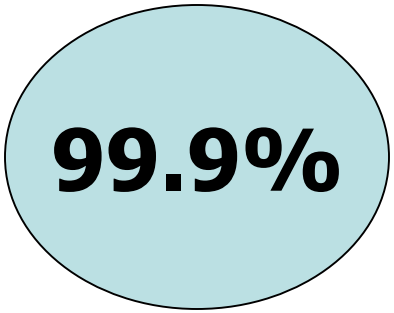
$$\text{Availability} = \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}}$$

- How to make availability approach 100%?

MTTF → infinity (high reliability)
MTTR → zero (fast recovery)

Subtleties

- Which has higher availability?
  - Two 4.5 hour outage / year
  - 1 mn outage / day



- For an Internet-base company such as EBay or AOL, which would be more desirable? Why?
- For an autonomous rover?

Safety

Absence of:

Catastrophic consequences on the users or the environment

- Are commercial aircraft “safe”?
- They crash very occasionally. How many crashes are too many?
- Are cars “safe”? They crash quite a lot

45K deaths/yr; 900/week = 2 fully loaded Boeing 747/week

Risk

- Risk is the expected loss per unit time

$$\text{Risk} = \sum \text{pr}(\text{accident}_i) \times \text{cost}(\text{accident}_i)$$

- *Safety* is expressed as an *acceptable* level of loss

Reliability vs. Availability vs. Safety

- They are not the same.....
- Example:

  A system that is turned off
  is not very reliable,
  is not very available,
  but is probably very safe
- In practice, safety often involves specific intervention

Confidentiality

Absence of:

Absence of unauthorized disclosure of information

- Microsoft source code vs. Linux source code
- Web browsing
- Operating Systems Security Model
  - Files, Memory
- Medical records
- Credit card transaction records
- School grades

Integrity

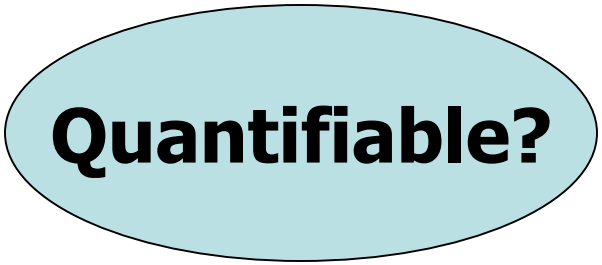
Absence of:

Absence of improper system state alterations

- Operating systems
  - Files, memory, network packets
- Linux kernel backdoor attempt
- Database records
- Your bank account
- File transfer
- Did I really get the right version of software XYZ?
- …

Security

- Security is a combination of attributes:
  - Integrity
  - Confidentiality
  - Availability



- Under different circumstances, these attributes are more or less important:
  - Denial of service is an availability issue
  - Exposure of information is a confidentiality issue

Maintainability

Ability to undergo repairs and modifications

- Maintenance
- Evolution
- Composition
- Manageability

KISS

(Keep It Simple Stupid)

Recap

- Software part of a system
- Facets of dependability
  - Reliability
  - Availability
  - Safety
  - Security (confidentiality, integrity)
  - Maintainability