1 Introduction - February 19, 2020

1.1 Defining Dependable Systems

QUOTES:

> *A distributed system is a system where a computer of which you did not know it exists can prevent you from getiing your job done.* - Leslie LAMPORT
>
> *There is perhaps a market for maybe five computers in the world.* - TJ WATSON

FAULT → ERROR → FAILURE
- Train delayed because of tree has fallen on the tracks
- Travelers reach destination too late
- Alice misses her exam

	FAULT	ERROR	FAILURE
Train:	Tree fallen	no train	delay for passengers
Journey:	Train delay	delay	reached destination 2h after intention
Exam:	arrival 2h late	missed time-slot	repeat exam

FAULT: cause of failure
ERROR: internal state of system, not according to specification
FAILURE: observable deviation of specification

FAULT examples:
- timing
- cables
- power supply
- messages lost
- data loss (solved with RAIDs)

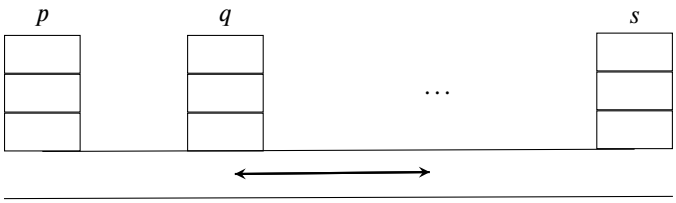
1.1.1 How to make systems tolerate faults

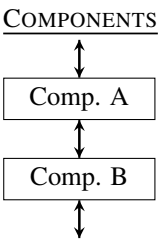
- PREVENTION
- TOLERANCE
  - Replication/Redundancy
  - Recovery
- REMOVAL
- FORECASTING/PREDICTION

SAFETY ≠ SECURITY
SAFETY is connected to loss of live/material due to accidents
SECURITY is connected to malicious intent

1.1.2 Defining distributed computation

Processes $\Pi = \{p, q, r, s \ldots\}$
$| \Pi | = N$





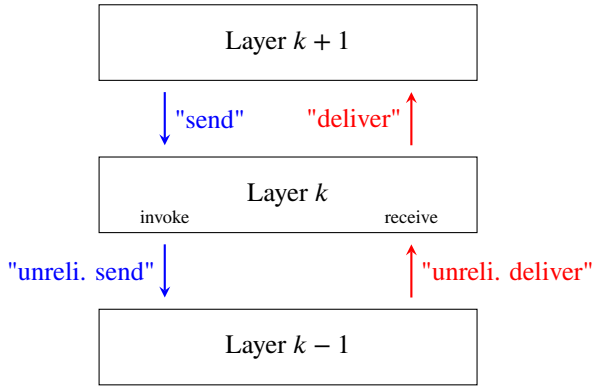
EVENTS for Component c:

$$\langle c, event \mid param_1, param_2 \ldots \rangle$$

upon $\langle c, ev_1 \mid param_1 \rangle$ do
  do something
  trigger $\langle b, domore \mid p \rangle$

upon $\langle b, domore \mid p \rangle$ do

1.1.3 Layered modules



Events either travel:
- upwards (red): indication
- downwards (blue): request

Events on a given layer may be:
- input events (IN)
- output events (OUT)

1.1.4 Module Jobhandler

Events:
  Request: ⟨*jh, handle* | *job*⟩
  Indication: ⟨*jh, confirm* | *job*⟩
Properties:
  Every job submitted for handling is eventually confirmed.

Implementation (synchronized) JOBHANDLER
State
  ...
upon ⟨*jh, handle* | *job*⟩ do
  "process job"
  trigger ⟨*jh, confirm* | *job*⟩

upon ...
upon ...

Implementation (asynchronized) JOBHANDLER
State
  $buf \leftarrow \emptyset$
upon ⟨*jh, handle* | *job*⟩ do
  $buf \leftarrow buf \cup \{job\}$
  trigger ⟨*jh, confirm* | *job*⟩

upon $buf \neq \emptyset$ do
  $job \leftarrow$ some element of buf
  "process job"
  $buf \leftarrow buf \setminus \{job\}$

1.2 Concurrency and Replication in Distributed Systems

2 2nd Lecture - February 20, 2020