A few ideas from distributed systems for PL folk

Lindsey Kuper (UC Santa Cruz)

lecture 3 agenda:

- safety & liveness recap; more examples
- fault models: the "two generals" thought experiment
- Brief highlights of a couple projects from my group:
  - Inductive diagrams for causal reasoning (COPSUL '21)
  - Library-level choreographic programming (ICFP '23, PLDI '25)

↳ interpreters ♡

safety: a "bad" thing doesn't happen

liveness: a "good" thing eventually happens

causal delivery (take 1):

if send(m_1) → send(m_2),

then for all processes P delivering both,

deliver$_P$(m_1) → deliver$_P$(m_2).

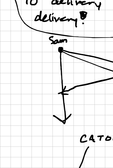
causal delivery (take 2):

if send(m_1) → send(m_2),

no process can deliver m_2 unless it has already delivered m_1.

totally-ordered delivery:

IF a process delivers m_1 and then m_2, then all processes delivering both deliver m_1 first.



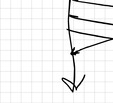
FIFO delivery (aka ordered delivery):

IF a process sends m_1 followed by m_2, then any process delivering both delivers m_1 first.

TO delivery doesn't imply FIFO delivery

let's talk liveness!

example: "The server eventually responds to every client request."



crash fault: a process fails by halting. (It stops sending/receiving messages.)

omission fault: a message is lost (a process fails to send or receive a message).

timing fault: n/a

Byzantine fault: a process behaves in an arbitrary or malicious way.

the "two generals" problem

Common knowledge of p:

- everyone knows p
- everyone knows that everyone knows p
- everyone knows ... etc.