LECTURE 4: REPRODUCING BUGS?

ANNOUNCEMENTS

- CHECKPOINT 1
    - MECHANICS
- FINAL PROJECT PROPOSALS? GROUPS?

TODAY

RECORD & REPLAY IN DISTRIBUTED PROGRAMS

∘ RECORD & REPLAY IN DISTRIBUTED PROGRAMS

→ WHY?

① Relates to two framing questions

↳ Obvious: How to use traces.

→ Less Obvious: Trace size: what to record.

② An old problem with no really good solutio

→ Why? Concurrency within processes & between processes.

③ Should, in theory, be more tractable today.

→ Why?

→ WHY USEFUL?

→ FRIDAY ATTEMPTS TO SHOW ONE REASON

→ BUT LET US BRIEFLY COMPARE TO PIVOT TRACING

- PIVOT TRACING GOAL

- REPLAY GOAL

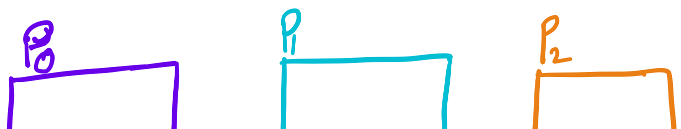
- HOW/WHY DOES PIVOT TRACING HELP?

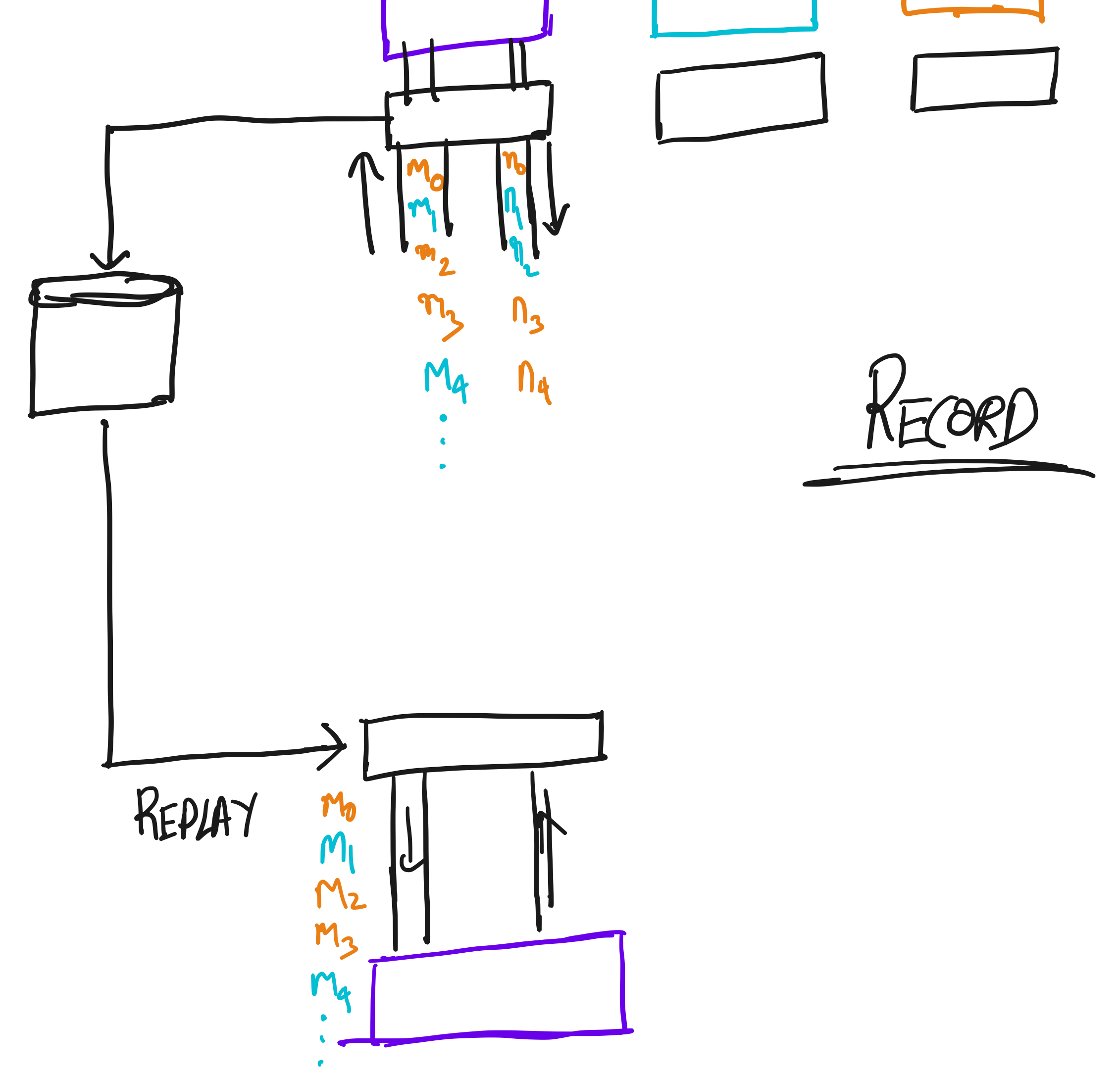
  Red. trace size

- WHEN IS PT BAD?

- CAN REPLAY HELP?

IDEALIZED RECORD + REPLAY



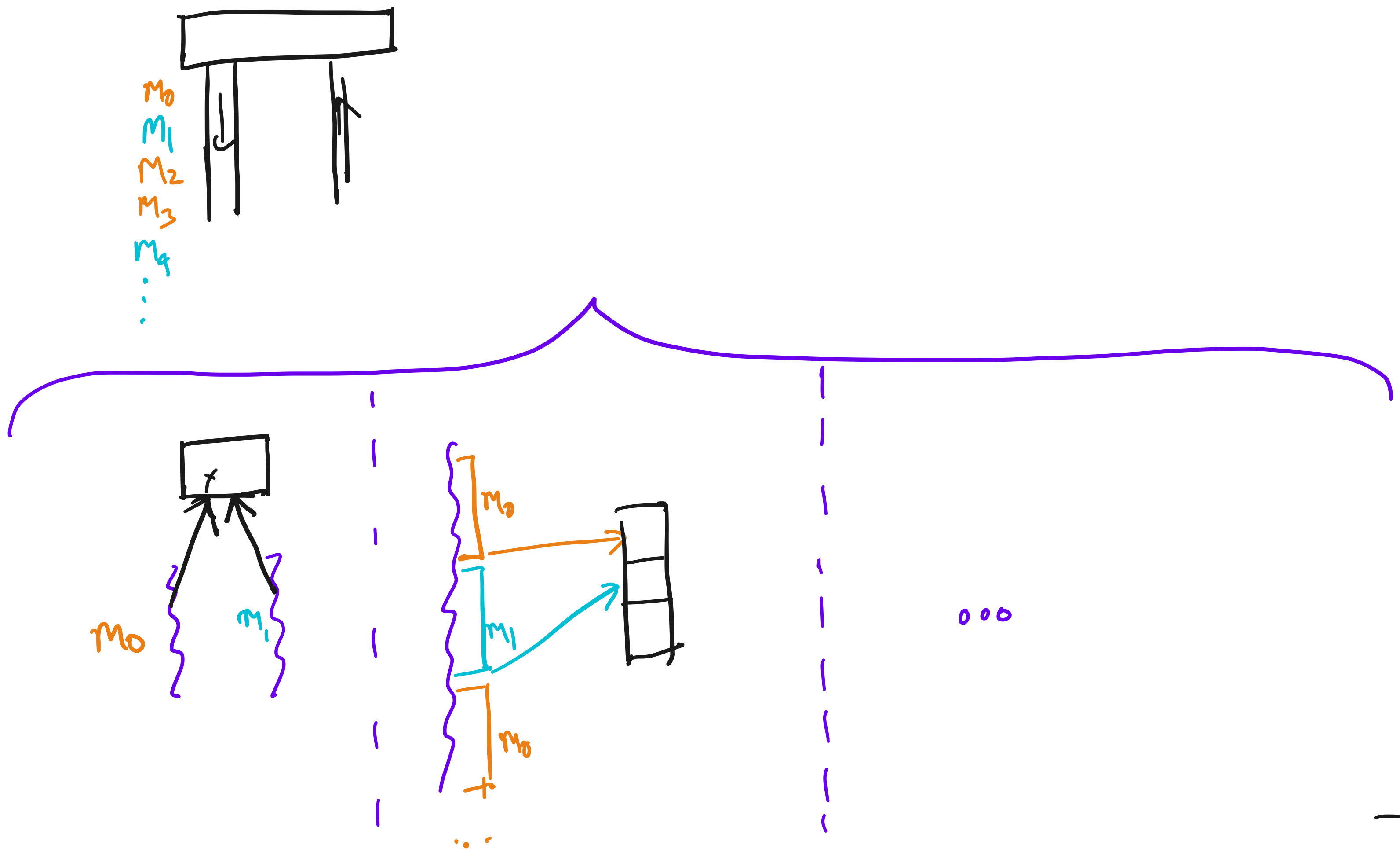


- WHEN CAN THIS IDEALIZED VERSION WORK?

<u>Assumptions</u> • Any state, any time

• Deterministic —

Problems With The Idealized Version



- How Much To Replay?

APPROACHES TO SOLVING THESE PROBLEMS?

- RECORD MORE:
    - Detect concurrent access & record operation order
    - During replay, don't allow concurrency & control order
    - Pros / Cons

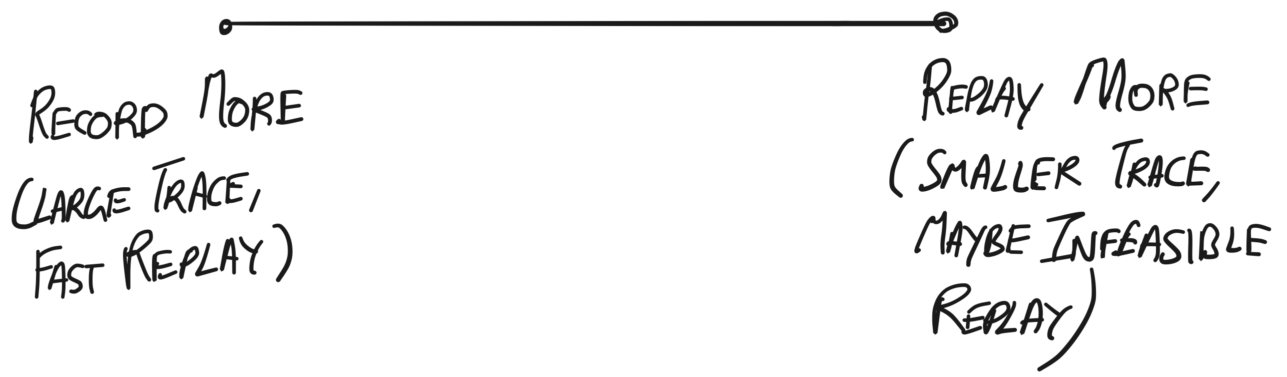
- Disallow concurrency

- Replay more
    - Try different orderings for concurrent/racing accesses.
        - ↳ Find one that produces the same state.
    - Problems
        - → Checking resulting state is the same?
        - → How do we know replay order ≡ original order?
        - → How do we even know all possible places where ordering matters?
        - → Replay time/complexity?

BOTTOM LINE: ALL APPROACHES NEED SOME DOMAIN ASSUMPTIONS

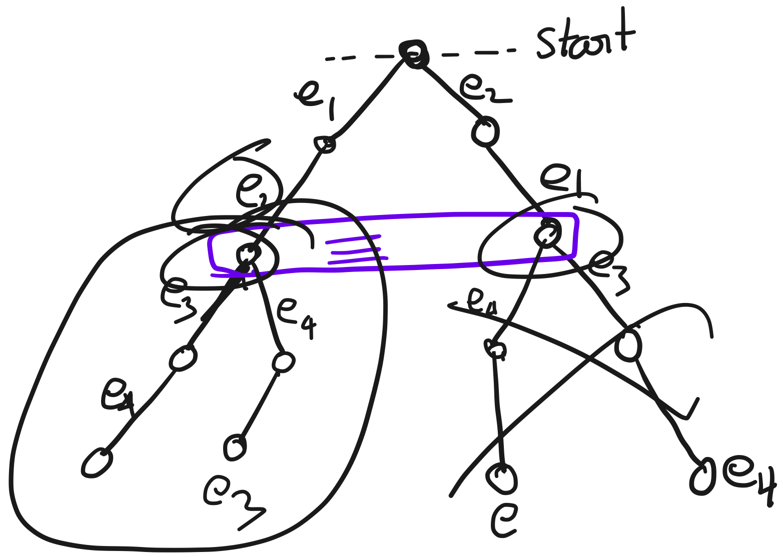
- WAY TO IDENTIFY CONCURRENCY
- WAY TO IDENTIFY EVENTS/EXECUTION POINTS WHERE ORDERING MIGHT MATTER
- WAY TO CHECK EQUIVALENCE
- ...

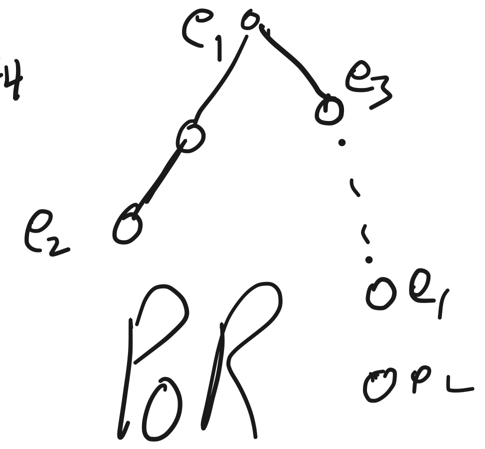
APPROACHES THUS FAR


RECORD MORE
(LARGE TRACE, FAST REPLAY)
REPLAY MORE
(SMALLER TRACE, MAYBE INFEASIBLE REPLAY)

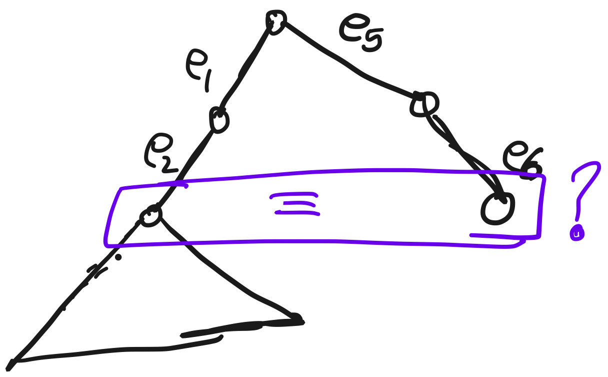
Can We Use Domain Knowledge to Reduce Replay Complexity?

- One Idea: Commutativity





More generally: Equivalence.

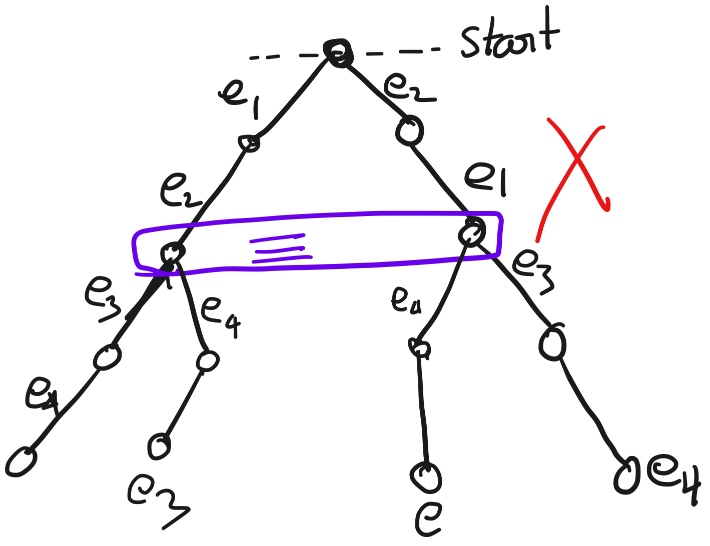


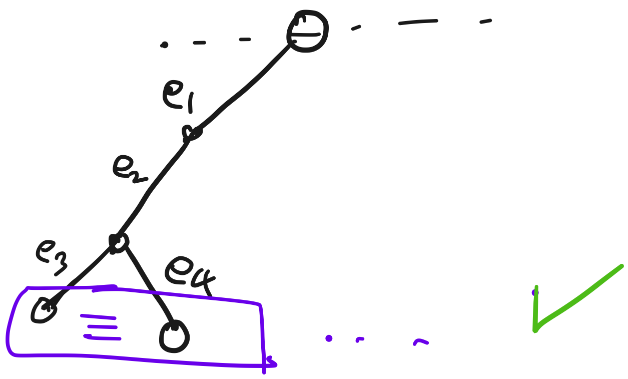
Challenge: State.

Why?

How to solve (maybe) state?

Depth First Search.





Limitations?

Fitting this all back into our setting.

- What we do have

- What we are missing

- Why there is some hope?
    - System calls & scheduler as sources of events
    - Data traces?
- Do we actually get savings on traces?

Limiting what we replay

Reminder: Why we have a problem?

What do we want ideally?

- Minimize number of events we have to replay.

Tools

↳ Checkpoints?

→ Minimization?

→

When do replay and minimization make sense?

- Correctness?
- Performance?
- Resource Exhaustion