Use clocks for ?

→ scheduling → marking points in time
→ durations / intervals of time

Clocks in computer

Time of day clock
→ tells you what time of day
→ can be synced btw machines using NTP

→ can jump forward / backward
→ leap second error Cloudflare Blog

Monotonic clock
→ only goes forward
→ good for duration
→ bad for marking time
→ NOT comparable between machines

Physical clocks:	Points in time	intervals / duration
Time of day clocks	MAYBE	NO
Monotic clocks	NO	YES

Logical Clocks

Only measure the order of events

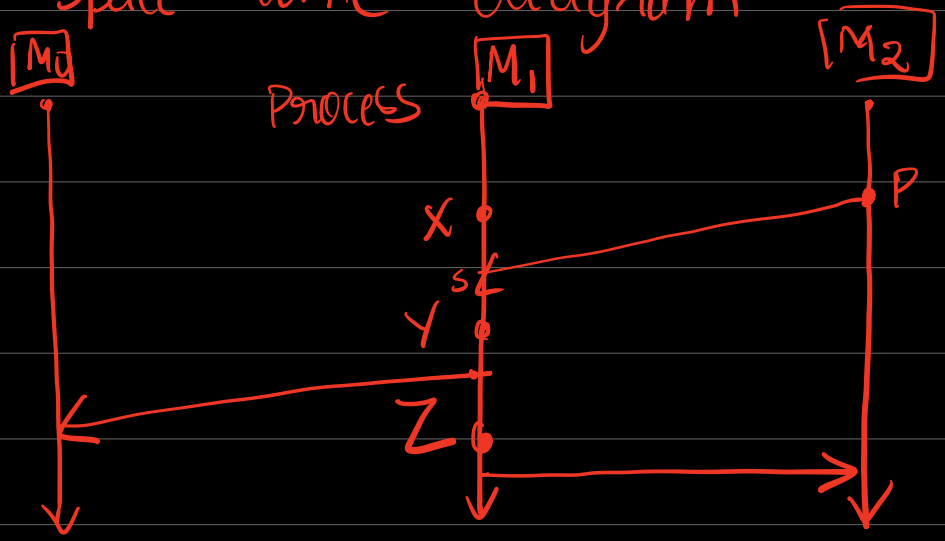
$A \rightarrow B$ 'A' happened before B

what does above tell about causality?

→ A could have caused B
→ B count not have caused A.

Lamport diagram

space-time diagram



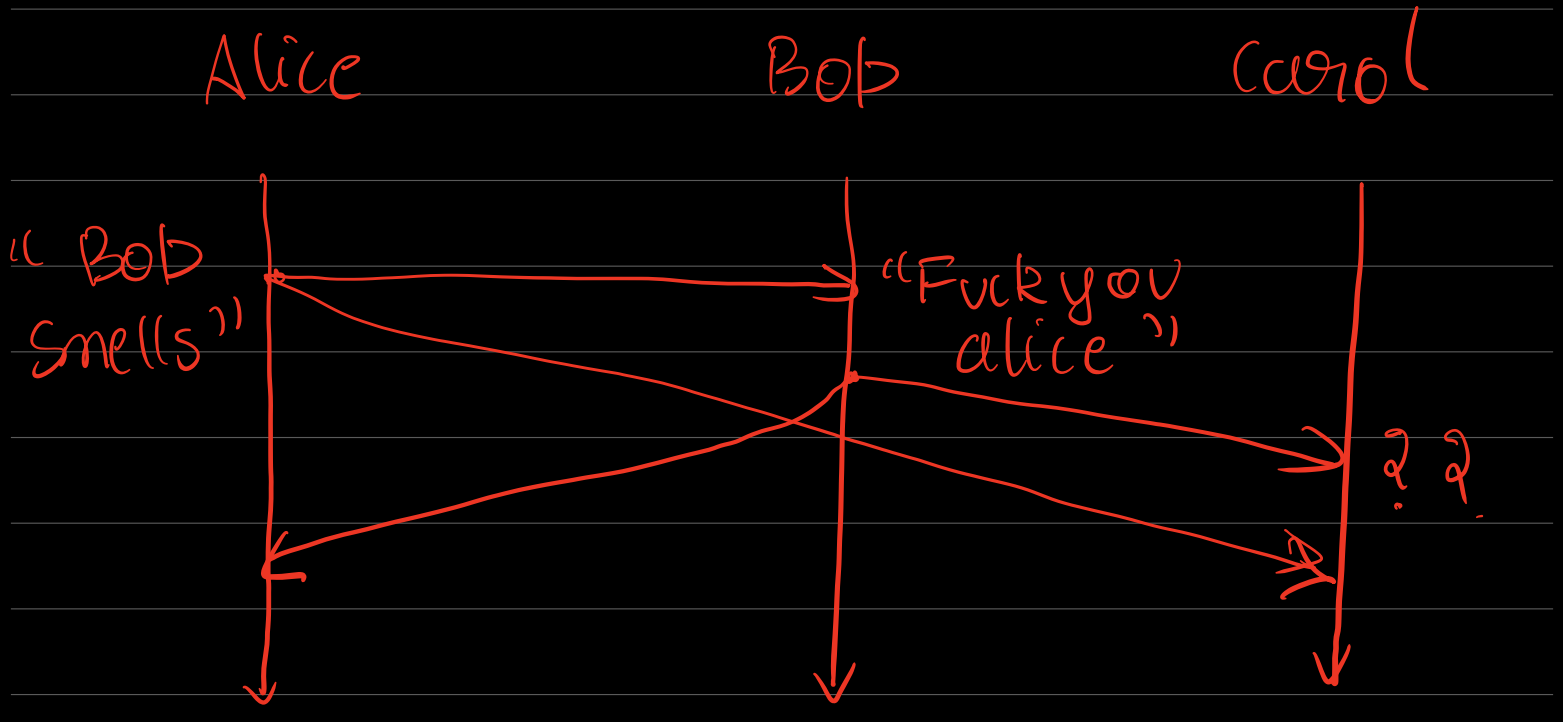
$X \rightarrow Z$
$Y \rightarrow Z$
$X \rightarrow Y$
$P \rightarrow S$

Given events A & B, we say $A \rightarrow B$ if:

- A & B occur on same process with A before B

- A is a send event & B is the corresponding receive event
- Transitivity: $A \rightarrow C$ & $C \rightarrow B$
  $\Rightarrow A \rightarrow B$

X, Y & Z are internal events
Events can be either send or receive.



?? → causal anomaly

why is the message delayed from alice to carol

→ Any reason: Slow, latency, etc.

Unbounded latency problem!

Network models

Synchronous Network

→ There exists an n such that no message takes longer than 'n' units of time to be delivered.

Asynchronous Network: More realistic approach.

→ There exists NO 'n' such that no message takes longer than 'n' units of time.

Partially Synchronous network.

→ There is an 'n', but we dont know what it is!

Happens before (→) relation.

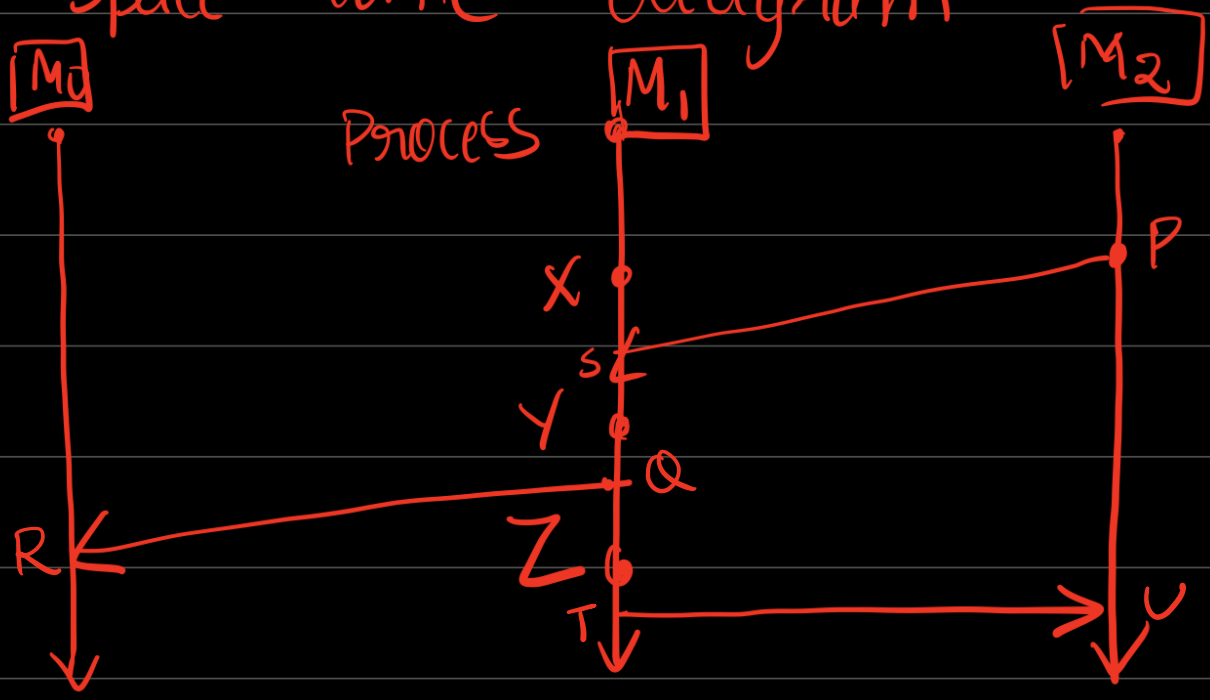
given events A & B:

→ if A & B occur on same process with A before B, then A → B

→ if A is a send event & B is the corresponding receive event

→ Transitivity: A → C and C → B then A → B

Space-time diagram



$X \to Z$

$Y \to Z$

$X \to Y$

$P \to S$

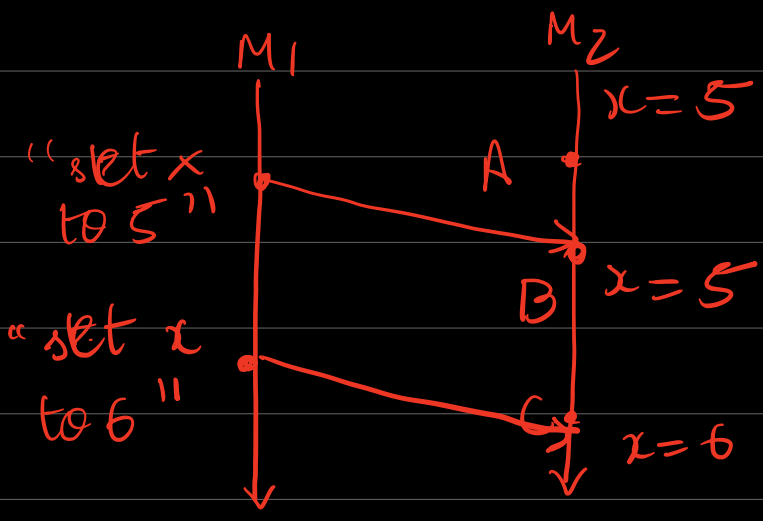
$Z \not\to R$

$R \not\to Z$

Z & R are concurrent

X & P are also concurrent

State and events



log

A ?

B → set x to 5

C → set x to 6

Argument :- Events also represent state!
Sequence of events can help create the current state