



ClockBound but Not Clock Limited: Making Time Accessible for Distributed Systems

Josh Levinson

(he/him)

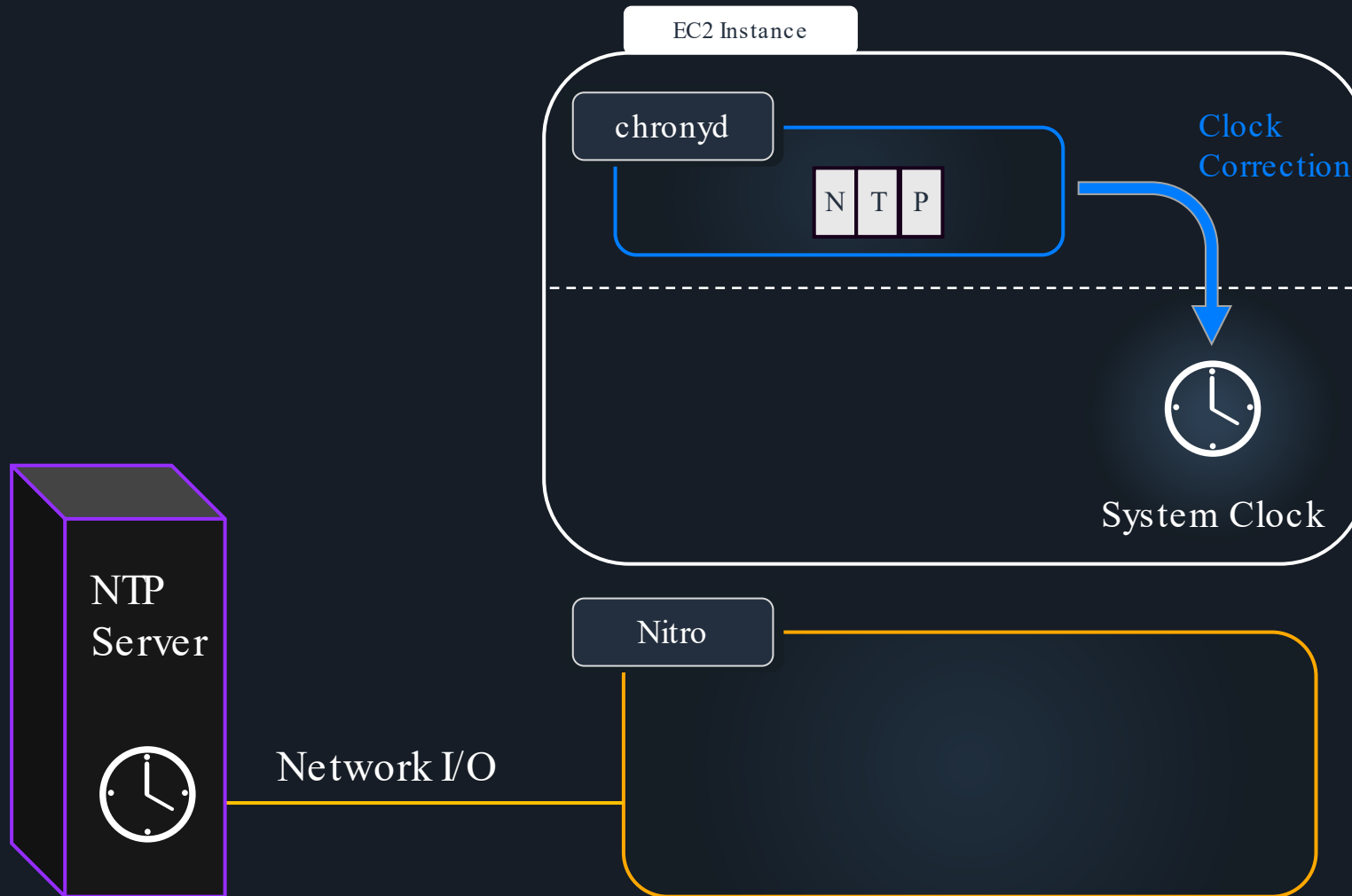
Principal Product Manager, Technical

Amazon Web Services

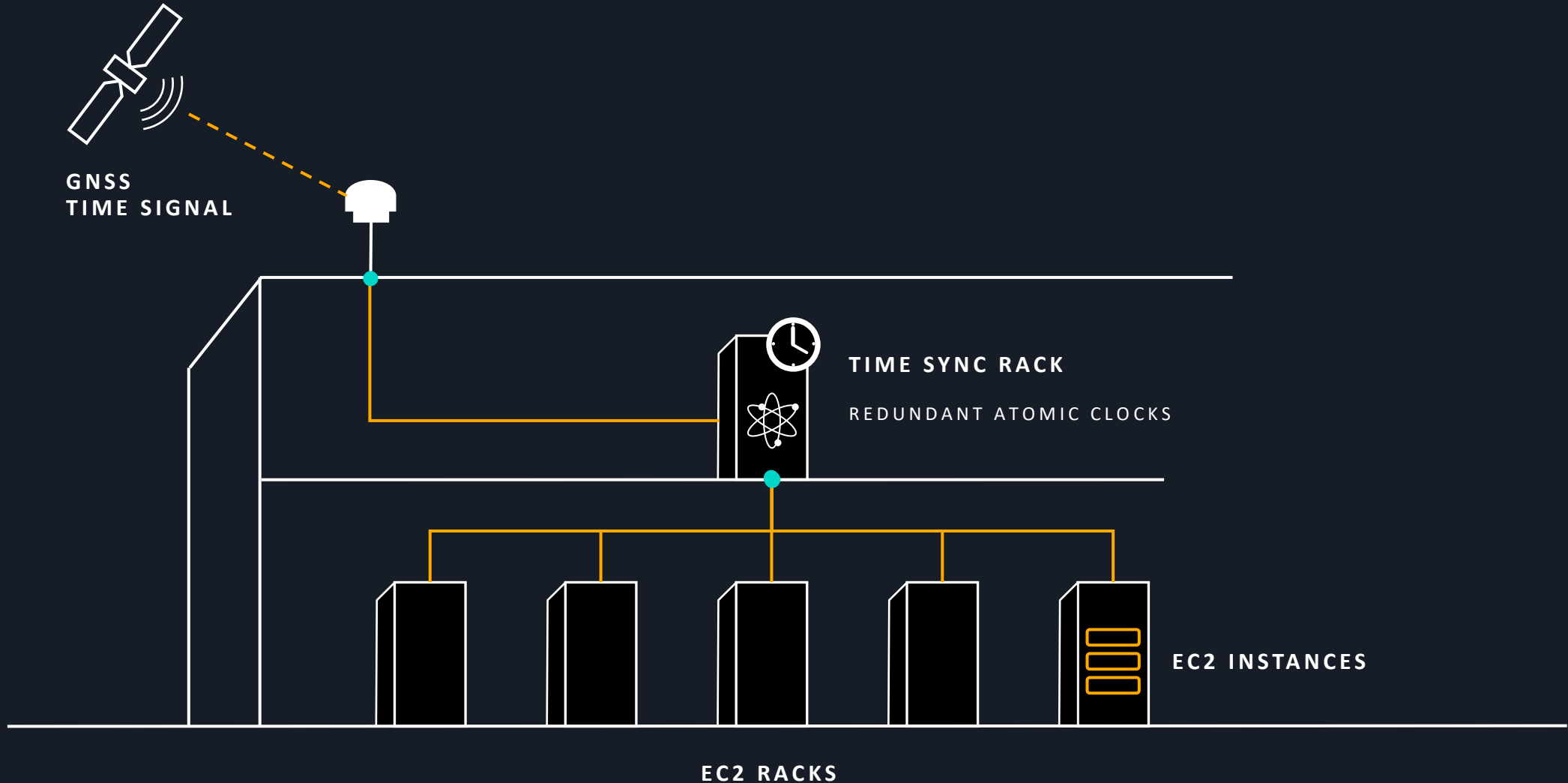
Agenda

1. AWS advances in clock sync
2. Leveraging good clocks globally

Amazon Time Sync circa 2017



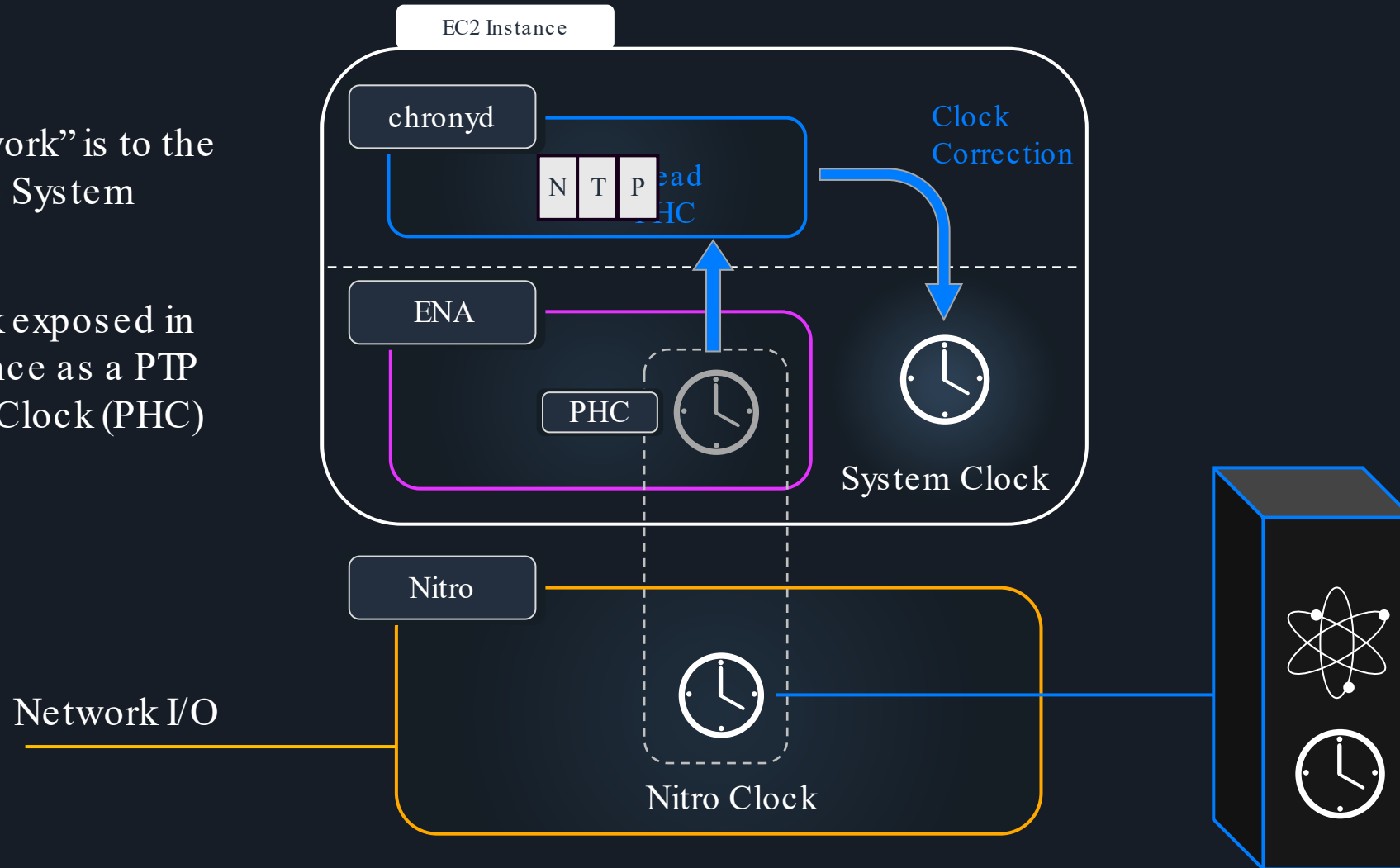
Dedicated Timing Infrastructure



Now: Per-Instance Hardware Clocks

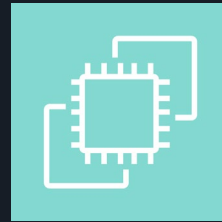
NTP “network” is to the local Nitro System

Nitro clock exposed in your instance as a PTP Hardware Clock (PHC)



Amazon Time Sync Precision Time Update

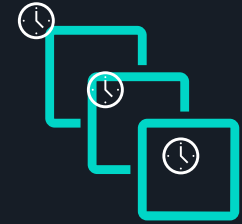
Launched 2023



Amazon Elastic
Compute Cloud
(Amazon EC2)



Hardware
reference



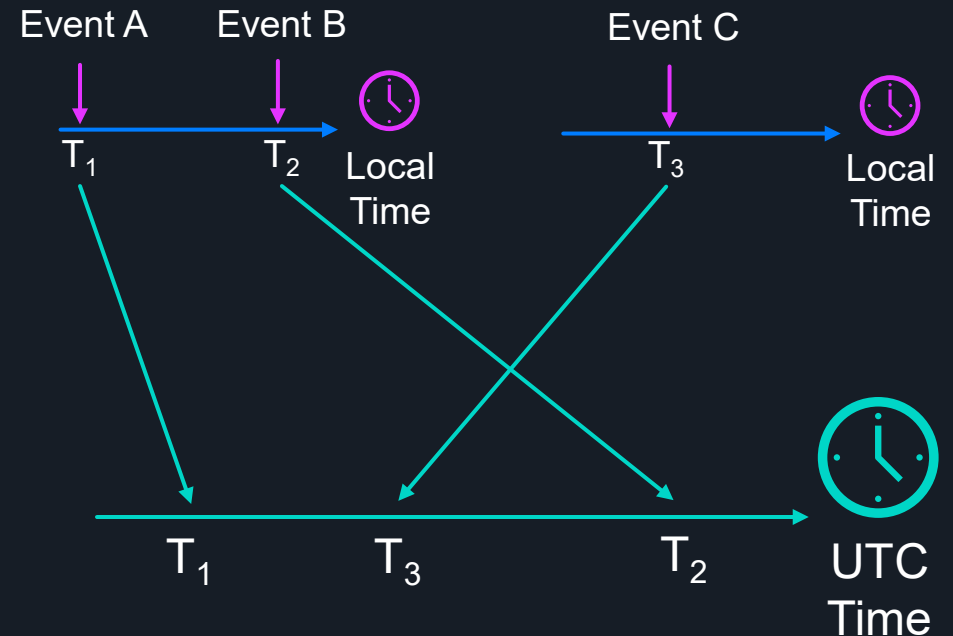
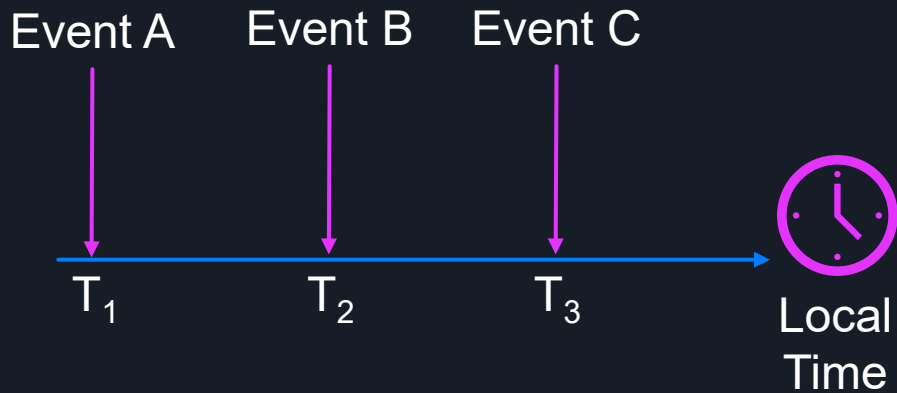
Every Amazon
EC2 instance
now has a
reference clock

- Built on the AWS Nitro System
- PTP Hardware Clocks or NTP
- North America, Europe, APAC

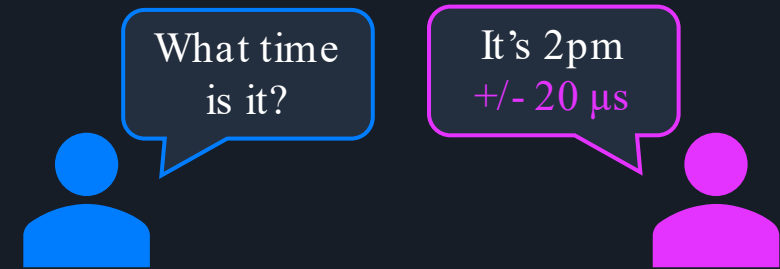
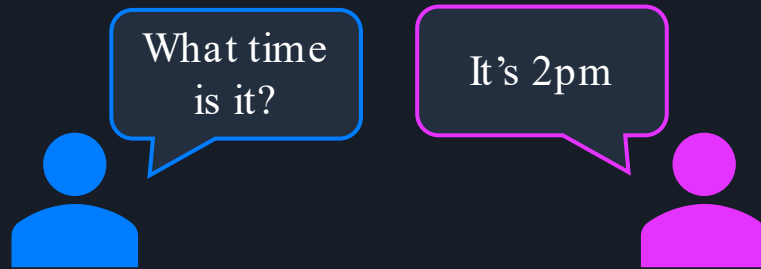
- 
1. Advances in clock sync
 2. Leveraging good clocks globally

Distributed systems have multiple clocks

- **Ordering** defines **consistency** and **isolation** level of a database
- Ordering on one computer is easy
- Ordering on many computers is costly and/or complex



Changing how you read a clock



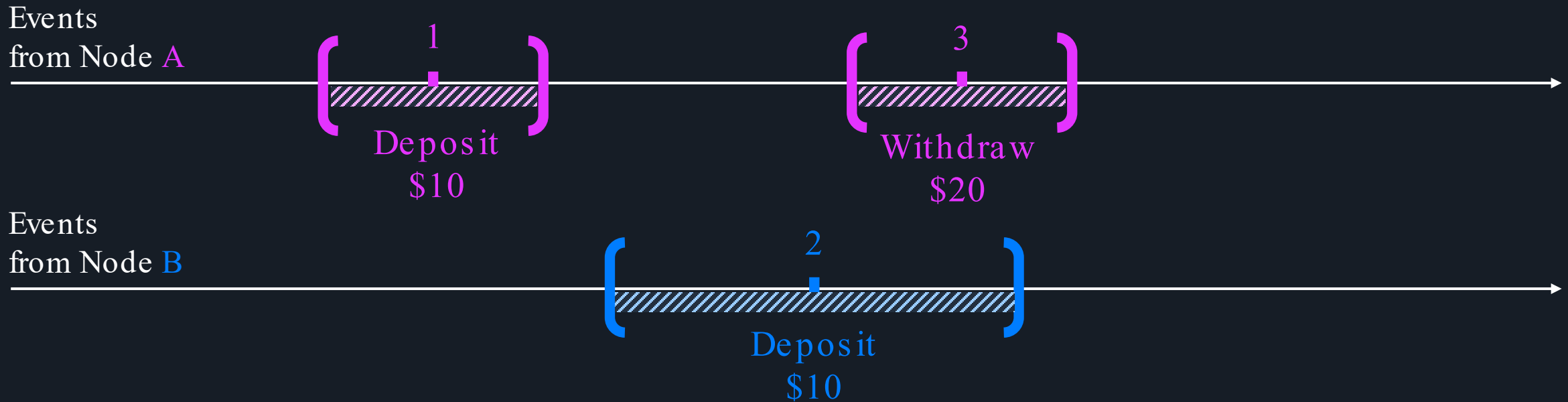
Perfect Time must be within the uncertainty window



Ordering clock bounded events

Using the uncertainty window, one can assert that:

- Event 1 is guaranteed to be in the past when compared to event 2 or 3
- Event 2 and 3 overlap uncertainty and must be processed carefully



Even good clocks have bounds

```
[ec2-user@xxx.xxx.xxx.xxx]$ chronyc sources
```

MS Name/IP address	Stratum	Poll	Reach	LastRx	Last sample
--------------------	---------	------	-------	--------	-------------

#* PHC0	0	0	377	0	-2ns[-4ns] +/- 32ns
---------	---	---	-----	---	----------------------

^- 169.254.169.123	1	4	377	13	+2728ns[+2949ns] +/- 90us
--------------------	---	---	-----	----	---------------------------

^- ec2-44-201-148-133.compu>	4	6	377	45	-26us[-26us] +/- 395us
------------------------------	---	---	-----	----	-------------------------

-2ns[-4ns] +/- 32ns

Nitro Time

Nitro NTP local

time.aws.com



ClockBound

Open-Sourced Since 2021

github.com/aws/clock-bound

In a single operation, you get:

- The current clock time
- The clock error bound
- The clock status

Supports PHC and NTP sources

Scales to the most demanding workloads

Written in Rust with clients available in Rust and C



Added December 2025: AWS Time Daemon



github.com/aws/clock-bound

Improved accuracy and reliability

Standalone time client and daemon

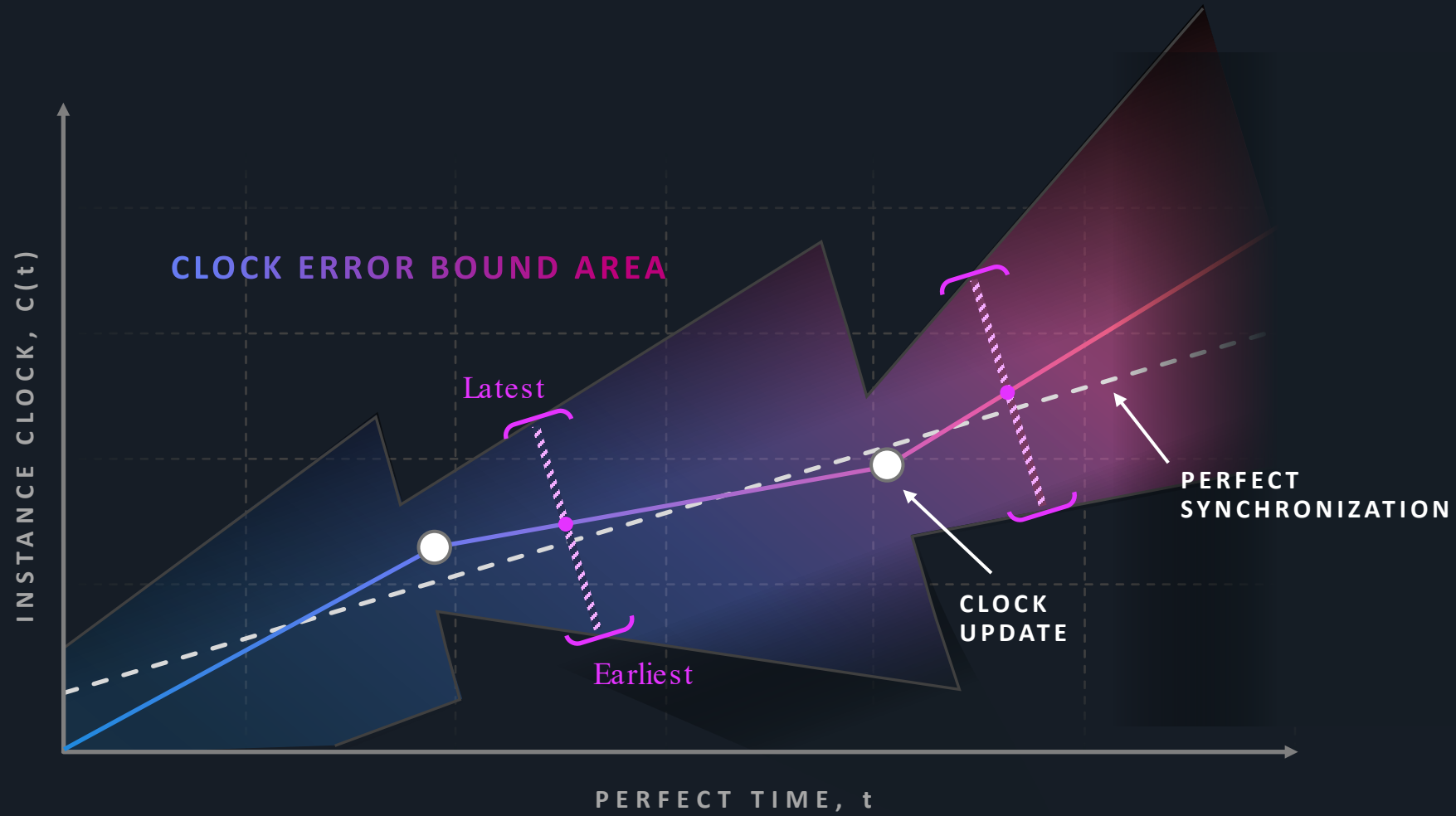
Built-in Metrics and Observability

Rust and shared memory design for performance

Cloud-first design

Simulator for you to test!

The Clock Error Bound



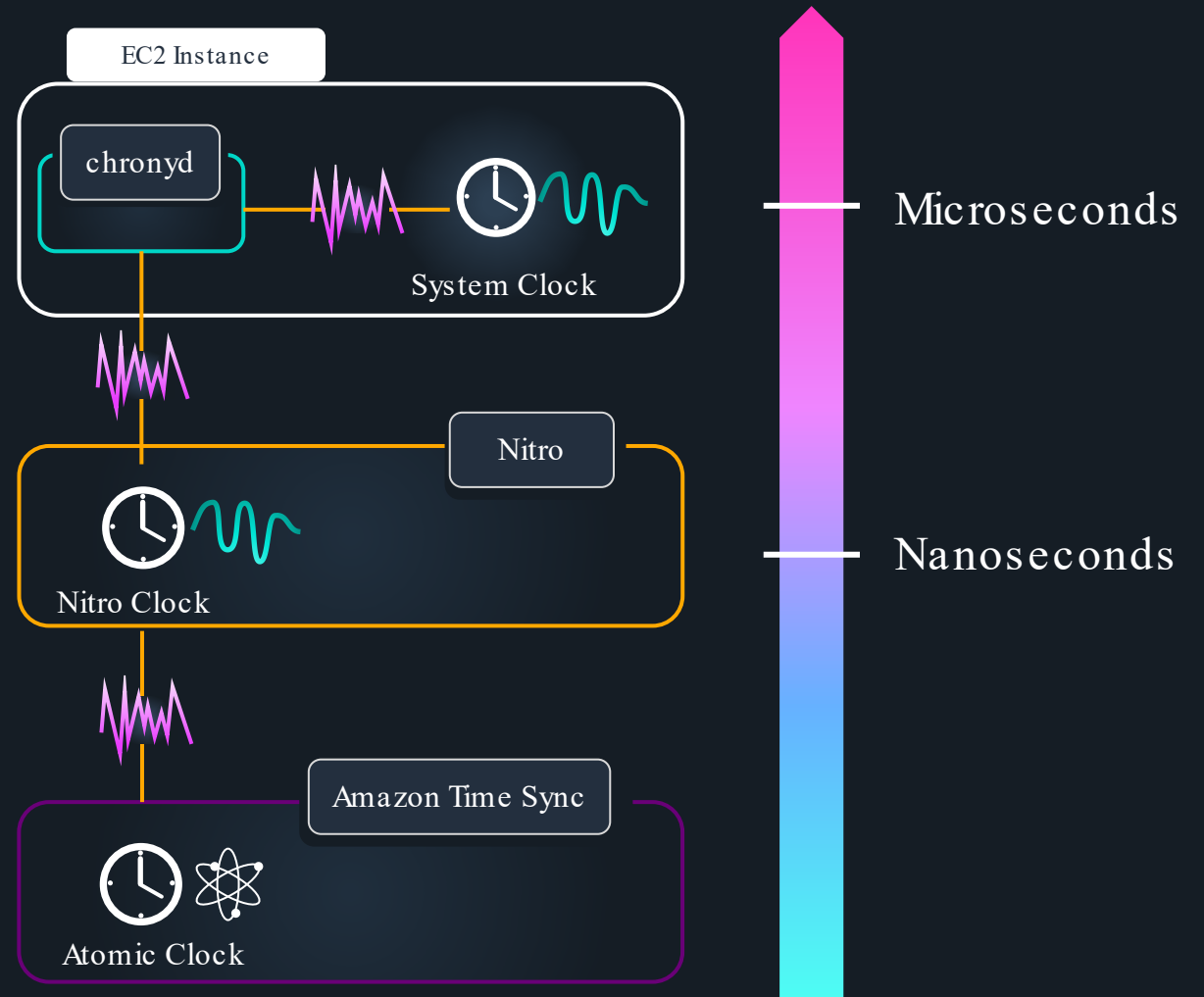
What makes the Clock Error Bound?

The tightest **Bound** ...

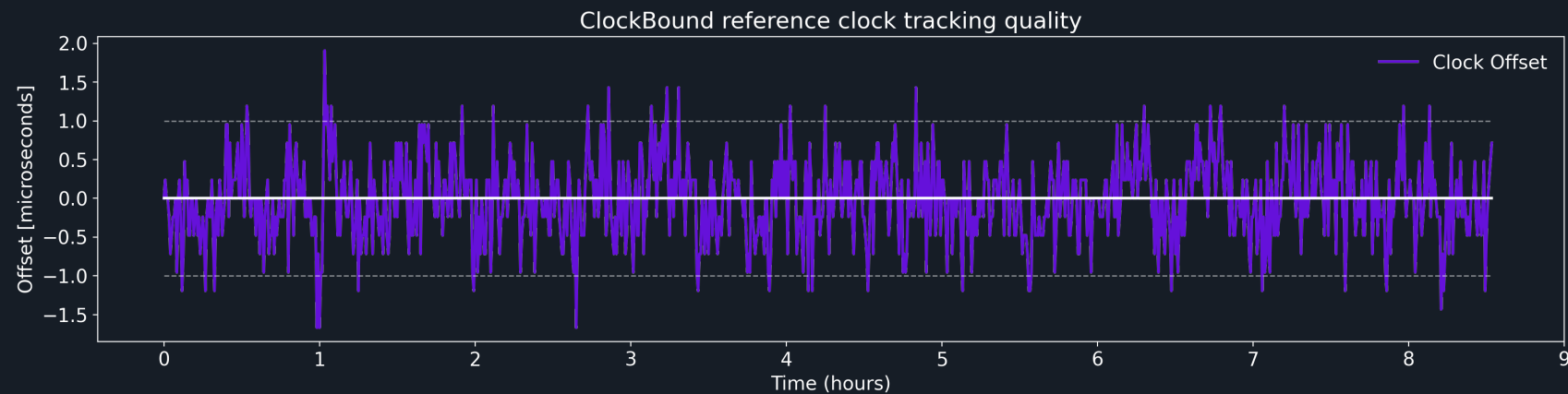
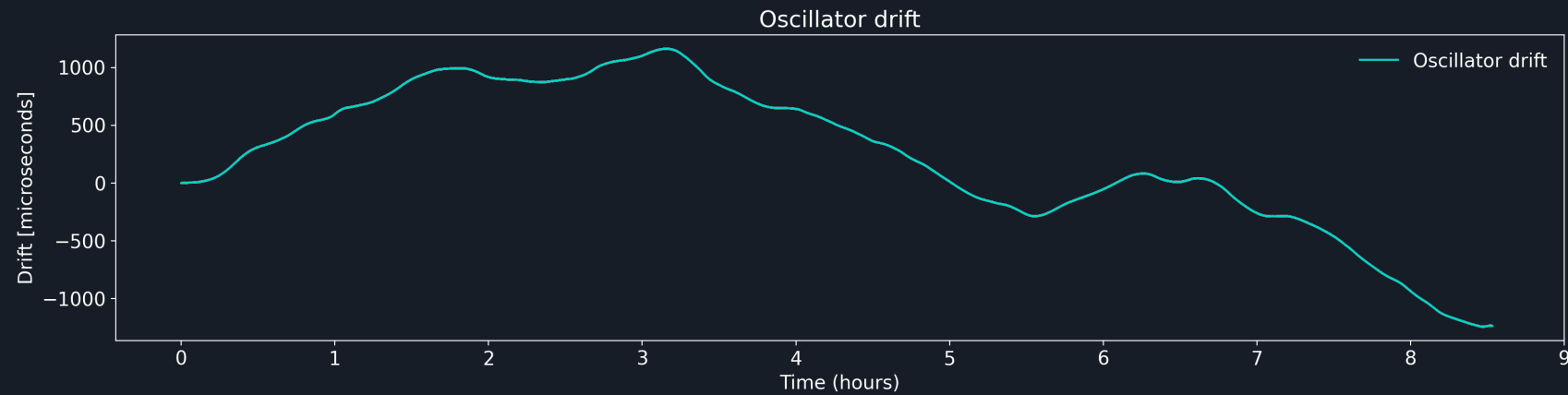
on the worst-case **Clock Error**...

due to oscillator drift ... 

and communication delays 



ClockBound tracking challenge



Clock Error Bound Improvements





Thank you

Josh Levinson

joshlev@amazon.com