Democratizing and Scaling Accurate Time for Distributed Applications

Josh Levinson

(he/him)
Principal Product Manager – Technical
Amazon Web Services

Julien Ridoux

(he/him)
Principal Software Engineer
Amazon Web Services

In this session, we will cover . . .

1. Clocks in distributed systems
2. Microsecond-accuracy on AWS (hardware clocks!)
3. Leveraging time in your workloads with Open-source ClockBound
4. Customer use case – YugabyteDB
5. Customer use case – Flow Traders
6. Amazon Aurora Limitless and Amazon Aurora DSQL
7. Demo

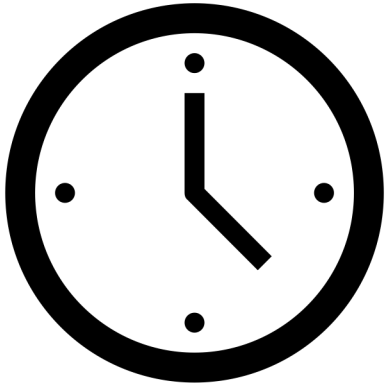
> The total ordering defined by the [logical clock] algorithm is somewhat arbitrary. It can produce anomalous behavior if it disagrees with the ordering perceived by the system's users. This can be prevented by the use of *properly synchronized physical clocks*.

Dr. Leslie B. Lamport

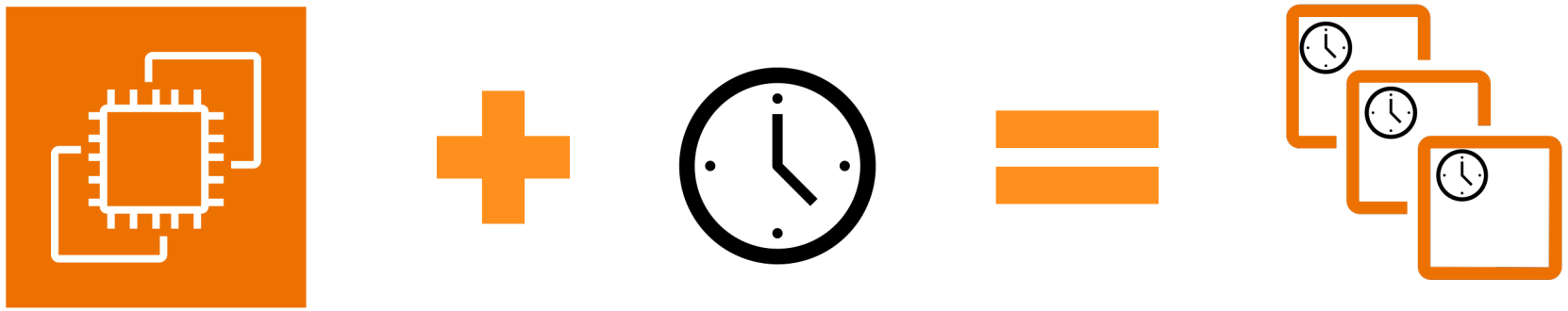
Time, Clocks, and the Ordering of Events in a Distributed System, 1978

Clocks as a primitive for improving any system

- The order of events in distributed computing is one of the most *fundamental and important problems in computer science*
- As the world gets more complex, trusted clocks are a simpler, streamlined solution – if your clocks are good enough!
  - One-way latency measurements
  - Consistent, point-in-time snapshots
  - Monitoring and observability
  - Time to live (TTL)
  - Ordering writes, reads, updates



Precision time on AWS



Amazon Elastic Compute Cloud (Amazon EC2)

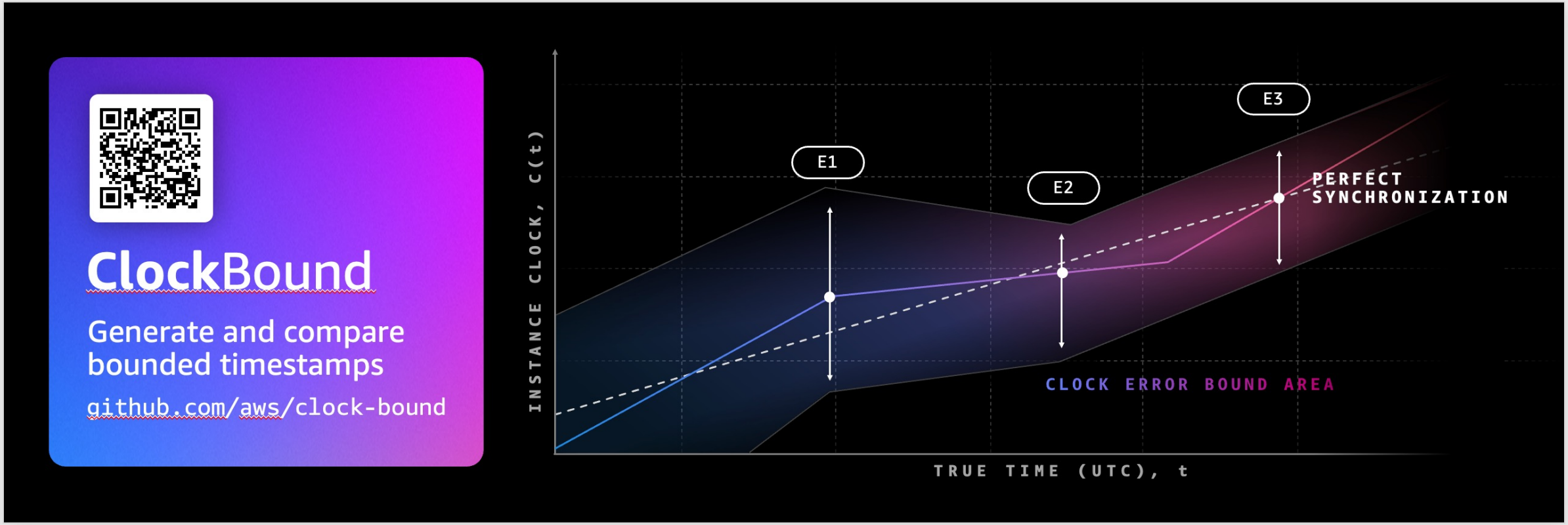
Hardware clock reference

Every Amazon EC2 instance now has a reference clock

- Amazon Time Sync Service added a 2023 precision time update – now everyone is Stratum 1!
- Built on the AWS Nitro System; supported in 6 AWS Regions, New York City Local Zone, and hundreds of Amazon EC2 instance types
  - First and only microsecond-range precision time service from any cloud provider
  - No additional cost for precision time on supported Amazon EC2 instances
  - Precision Time Protocol (PTP) hardware clock with ~20 µs in-guest clock error bound
  - Network Time Protocol (NTP) also available at ~80 µs in-guest clock error bound

Dr. Werner Vogels talking time at re:Invent 2024

Leveraging time in your workloads with ClockBound



- Open-source project to calculate clock error bounded timestamps
- Reliable method to compare and order timestamps of events across servers

YugabyteDB Unlocked performance with microsecond-accuracy

Before (max skew ~500 ms)

YugabyteDB uses a *hybrid logical clock* with maximum allowed clock skew upper bound of 500 ms

After adding the Amazon Time Sync Service

YugabyteDB implements a *Dynamic Hybrid Logical Clock* by fetching the maximum clock error on Amazon EC2 instances

Reduced latency

3x

improvement in transaction latency*

Increased throughput

2x

improvement in throughput*

Fewer retries

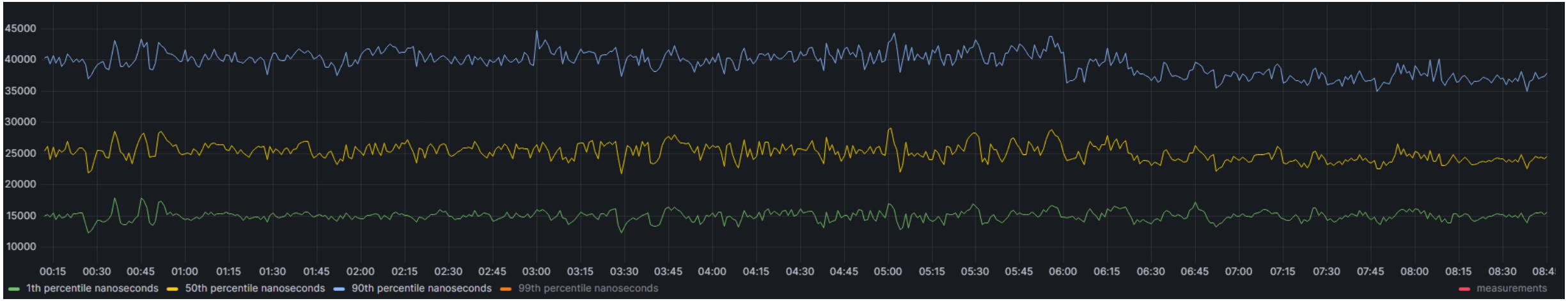
1,000x

improvement in transaction retries*

*Tests conducted using a simulated analytical workload: yb-sample-apps

FLOW TRADERS

Gaining the microsecond advantage



- In the high-frequency trading (HFT) industry, every microsecond counts!
- Flow Traders's previous time sources incorrectly measured network travel time, with measurements ranging from –56 µs to +169 µs
- After Flow Traders changed to the Amazon Time Sync Service, drift was consistently ~25 µs for the 50th percentile

AWS blog with

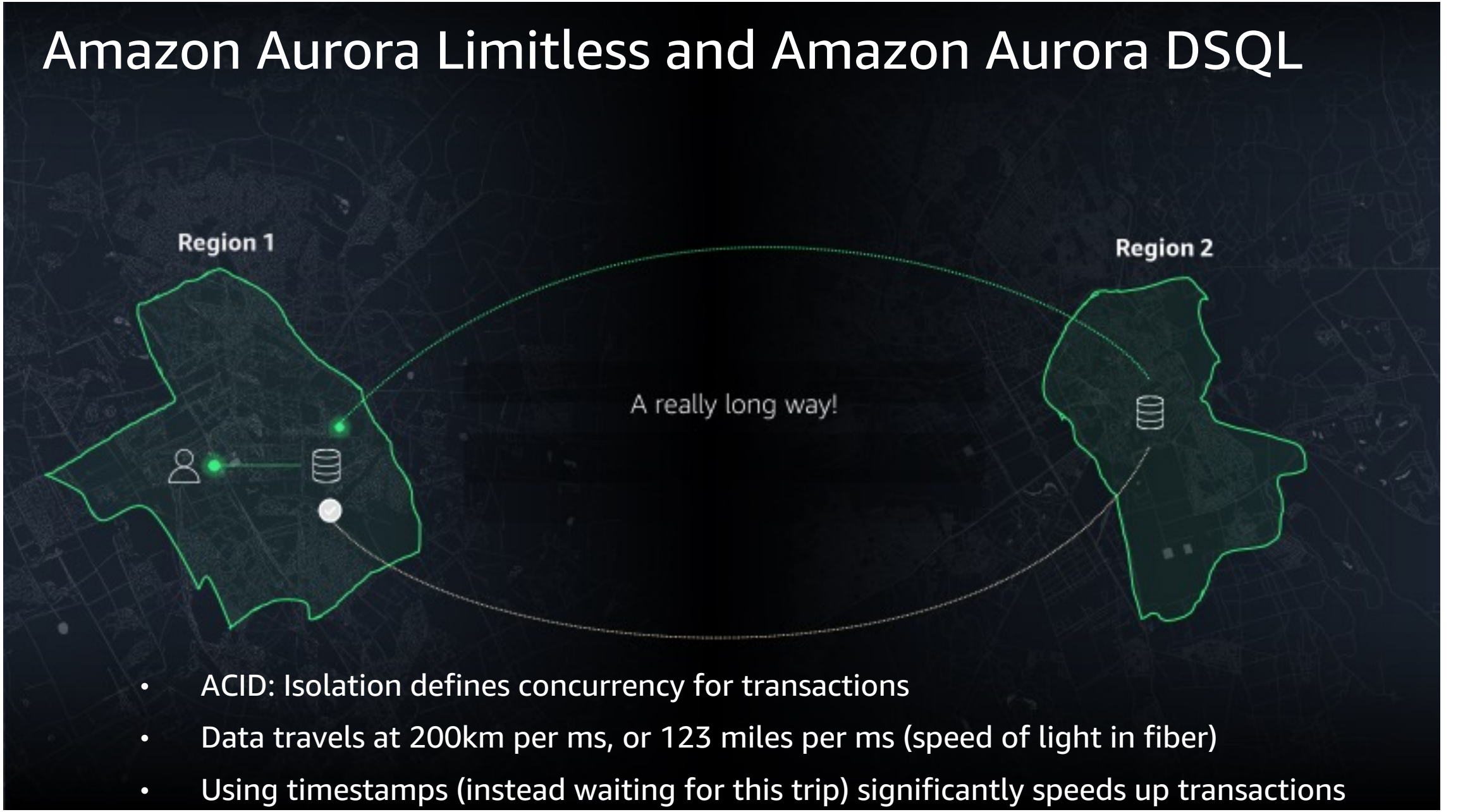
FLOW TRADERS

> For Aurora DSQL, we use high-precision physical clocks, which allows us to synchronize time with microsecond accuracy anywhere in the world. This high precision means we know the sequence of every single read and write with no additional latency.

Matt Garman

CEO, Amazon Web Services

Amazon Aurora Limitless and Amazon Aurora DSQL



- ACID: Isolation defines concurrency for transactions
- Data travels at 200km per ms, or 123 miles per ms (speed of light in fiber)
- Using timestamps (instead waiting for this trip) significantly speeds up transactions

Demo with Open-sourced ClockBound

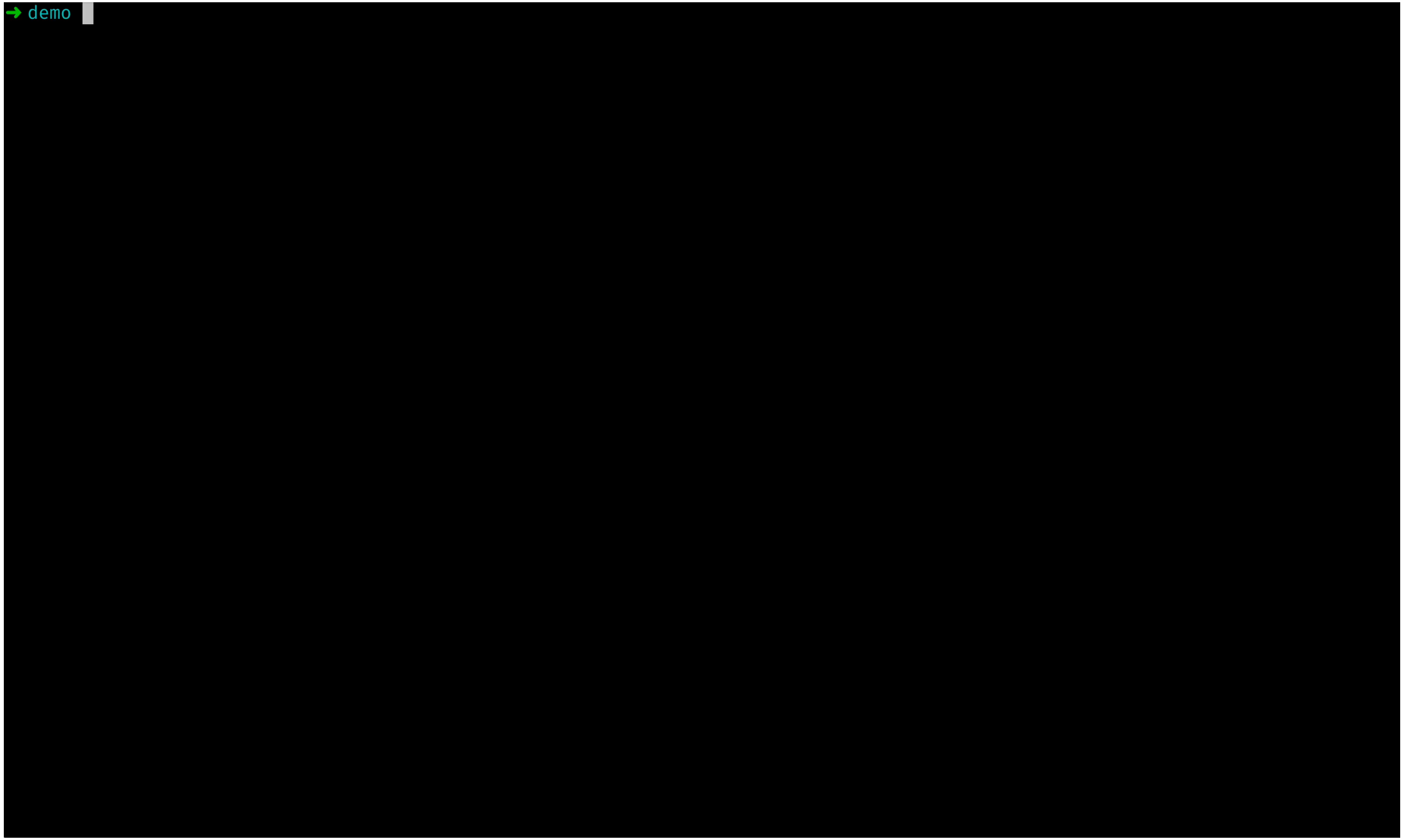
Use Ping to Measure network delays

```
❯ ping amazon.com
PING amazon.com (205.251.242.103): 56 data bytes
64 bytes from 205.251.242.103: icmp_seq=0 ttl=241 time=92.912 ms
64 bytes from 205.251.242.103: icmp_seq=1 ttl=241 time=89.673 ms
```

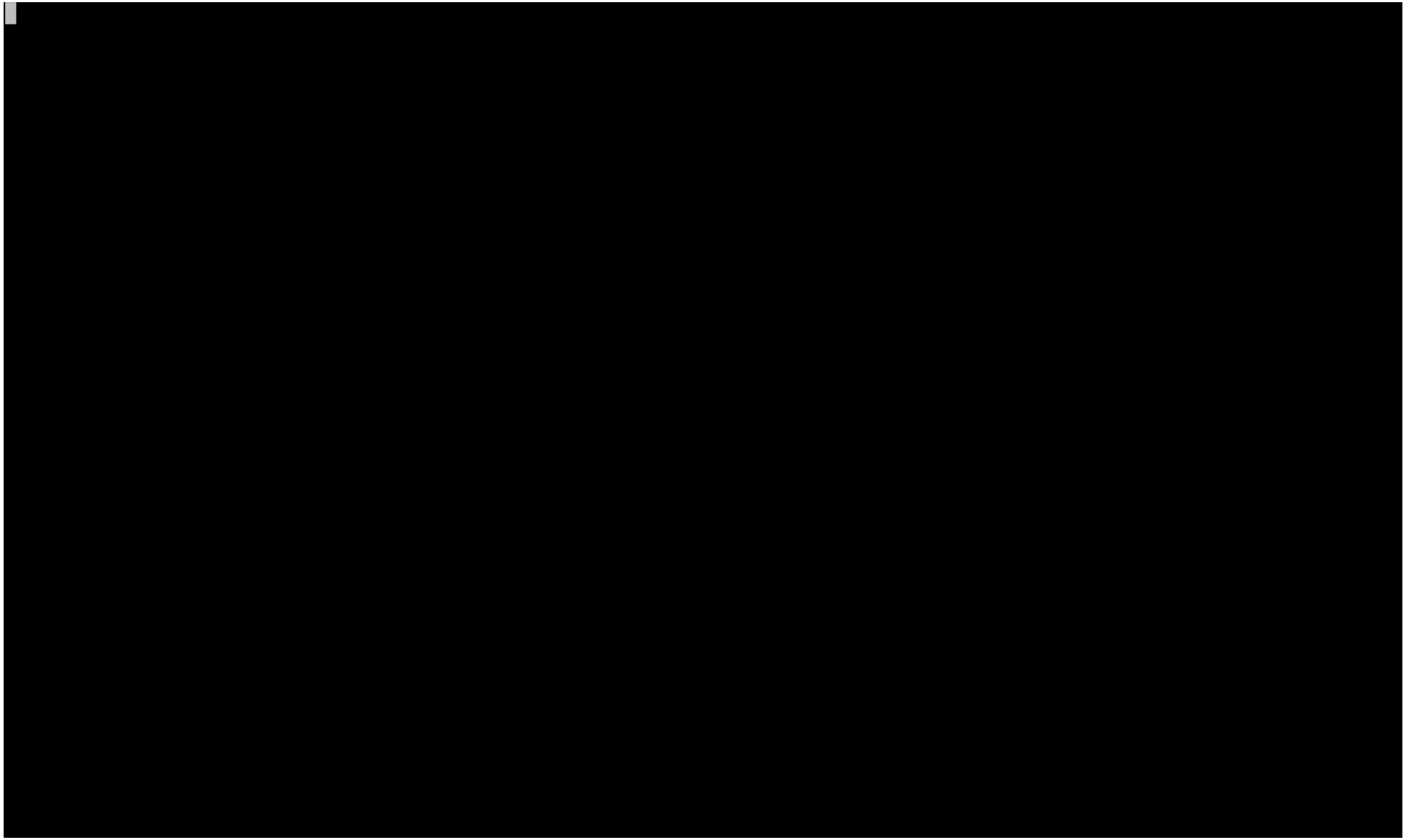
Use ClockBound to get a guarantee on clock accuracy on every timestamp

```
❯ clockbound-now
2024-11-22T16:23:00.788322 [+/- 58 microsecond]
```

Demo Video – Amazon Time Sync and ClockBound



Demo Video – Clocks and Network



How You Can Democratize Accurate Time

- Precision Time is an important primitive in today's and tomorrow's technology
- Trust and transparency in clock accuracy is paramount for distributed systems
- Think about where your customers use time/timestamps; not just at the hardware layer, but where customers build in applications
- Timing infrastructure resiliency is more critical than ever - customer applications are only as good as their worst clock!

Thank you!

Josh Levinson

joshlev@amazon.com

@notJustJoshin

linkedin.com/in/joshuablevinson

Julien Ridoux

ridouxj@amazon.com

linkedin.com/in/jridoux