An ML-based Approach to Clock Sync at Scale

Balaji Prabhakar

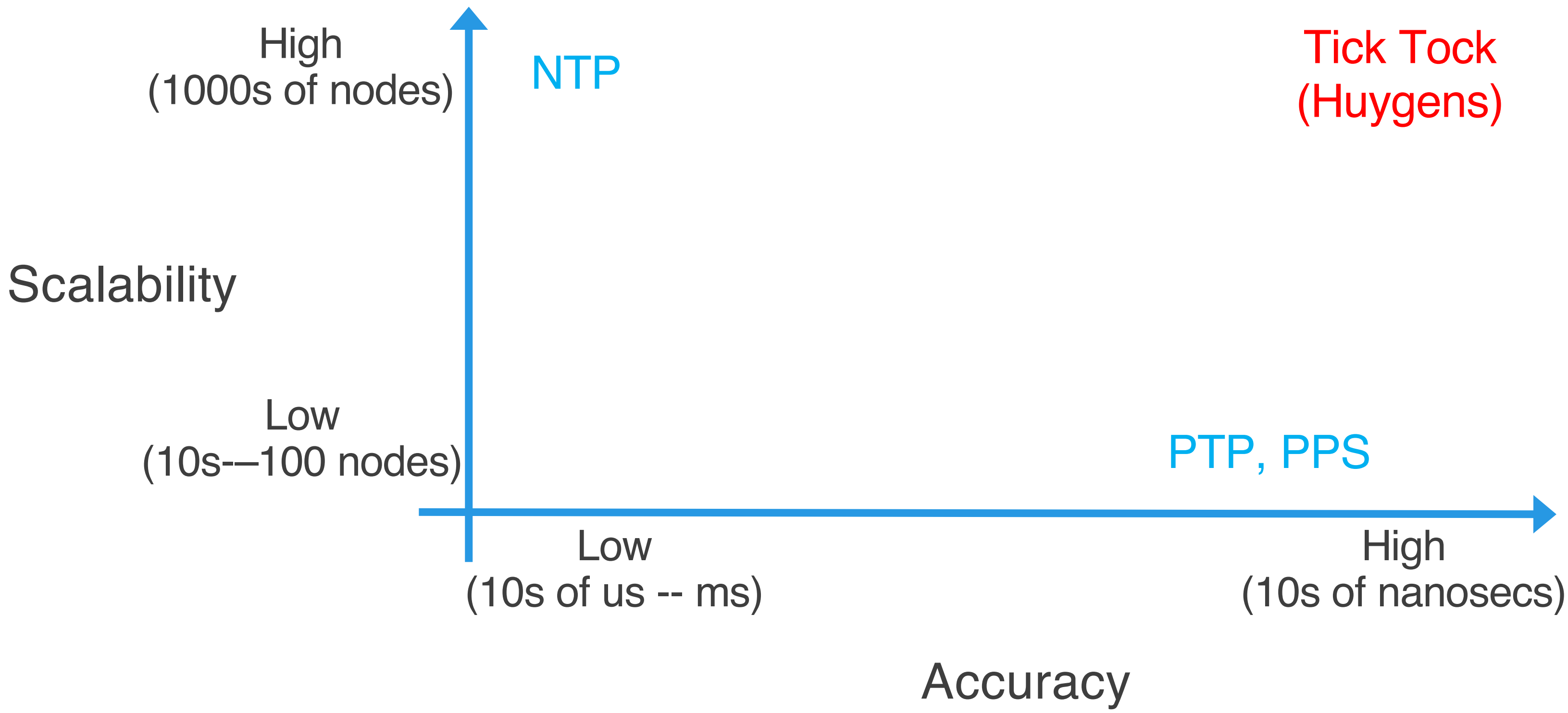
STAC Summit
London, Nov 2018

It's about time!

Accurately synchronized clocks are foundational
for any real-time distributed system

It's mission-critical for the financial services industry

Landscape of Clock Sync Solutions



Our Solution: High-precision, Software-based Network Clock Synchronization System

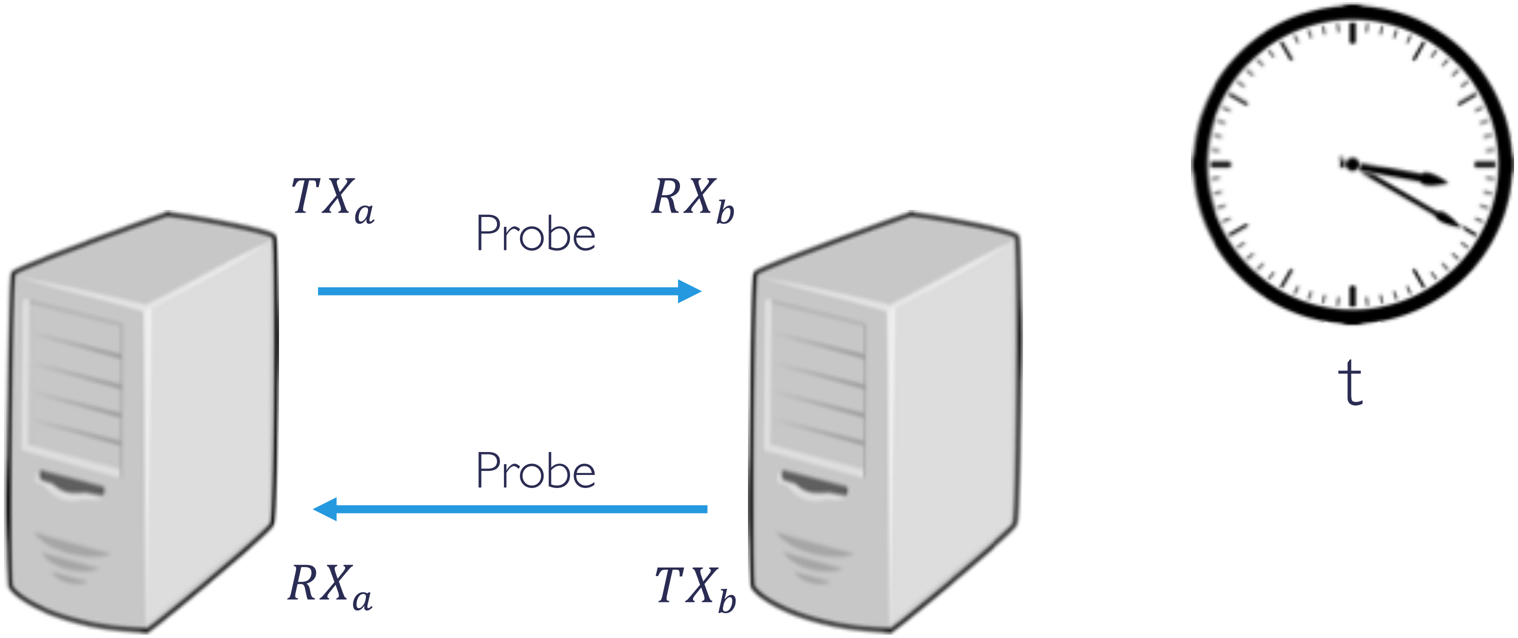
- **Nanosecond accuracy using NIC clocks. Tested at various data centers.**
  - Google 40G testbed: 40 racks, 5 stages of switching
  - Various financial data centers (Nasdaq, Deutsche Börse, etc)
  - Stanford 1G data center: 8 racks, 128 servers, 2 stages of switching
- **Scalable**
  - Up to tens of thousands of servers in on-premise data centers and public clouds
- **Synchronizes a variety of different clocks**
  - NIC clocks using hardware timestamps ➔ nanosecond-level accuracy
  - CPU clocks using software timestamps ➔ microsecond-level accuracy
  - Fast convergence, 4 - 6 seconds to achieve time sync
- **Easy to deploy:**
  - Drop-in replacement for NTP; no special purpose hardware required; set up in 30 minutes

The New York Times

" Time Split to the Nanosecond Is Precisely What Wall Street Wants"

https://www.nytimes.com/2018/06/29/technology/computer-networks-speed-nasdaq.html

Syncing Clocks with Probes



$t_a = t + \Delta t_a$

$t_b = t + \Delta t_b$

Basic Idea behind Huygens

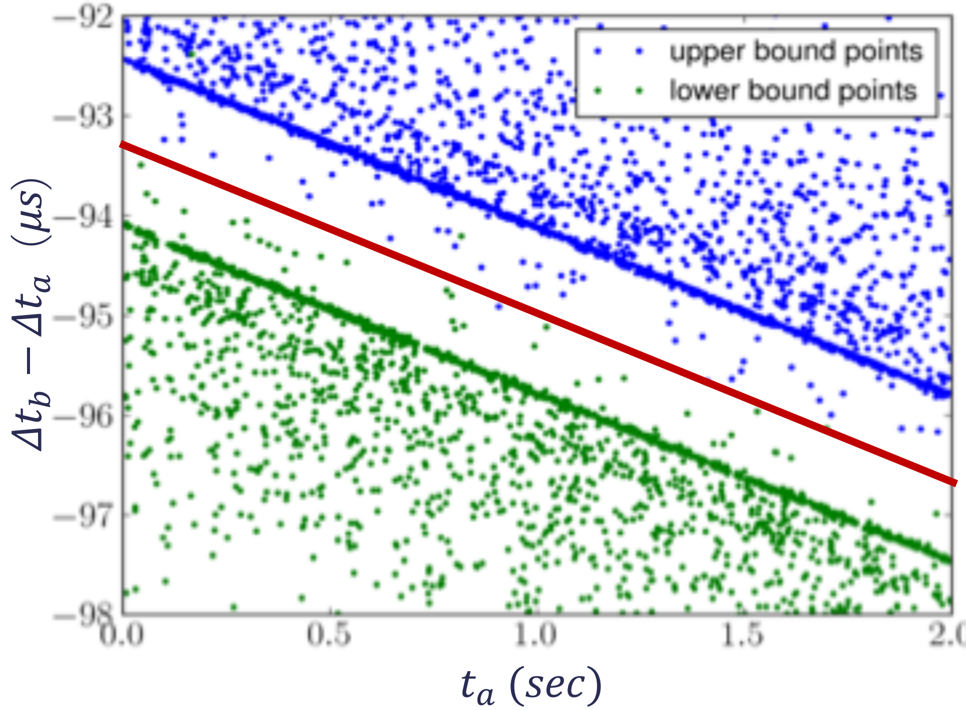
Probe from A to B

- Receive time = transmit time + delay
- $RX_b - \Delta t_b = TX_a - \Delta t_a + Propogation\ and\ queueing\ delay$
- $\Delta t_b - \Delta t_a = RX_b - TX_a - Propogation\ and\ queueing\ delay$
- $\Delta t_b - \Delta t_a < RX_b - TX_a$

Probe from B to A

- $\Delta t_b - \Delta t_a > TX_b - RX_a$

Clock Bounds Over Time



3 Key Ideas for Finding the Red Middle Line

- **Support Vector Machines**
  - ML technique for classification of labeled data
- **Coded probes**
  - A method of detecting and rejecting “impure” probe data
- **Network effect**
  - Key step: ensures clocks are not just pairwise synchronized, but also “transitively synchronized”

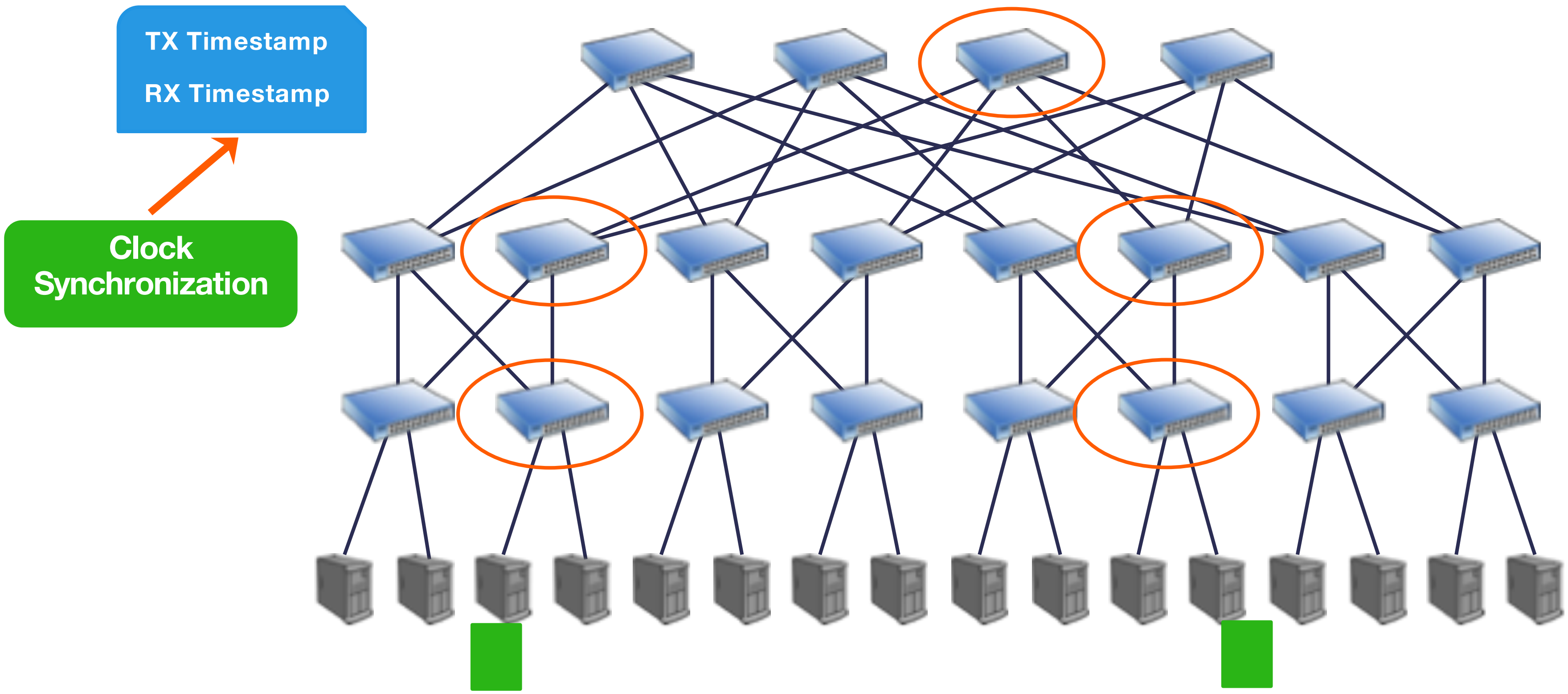
Details + Talk in our NSDI’18 Paper:

”Exploiting a Natural Network Effect for Scalable, Fine-grained Clock Synchronization”

https://www.usenix.org/system/files/conference/nsdi18/nsdi18-geng.pdf

Network Telemetry From The Edge: Tomography

From total time in the network, determine time spent in each switch



Thank You!

Questions? Please contact:

Balaji Prabhakar, balaji@ticktocknetworks.com
Dan Zheng, dan@ticktocknetworks.com

NSDI'18 Paper:

"Exploiting a Natural Network Effect for Scalable, Fine-grained Clock Synchronization"

https://www.usenix.org/system/files/conference/nsdi18/nsdi18-geng.pdf