Timeliness: A new paradigm for Distributed System and Networking

Balaji Prabhakar and Mendel Rosenblum
Self-Programming Networks Research Group
Stanford University
Major Trends in the Evolution of Cloud Computing

2000—2009: Virtualization
- Server virtualization
- Network virtualization
→ Enabled cloud computing revolution

2010—2019: Data Storage/Processing Tech
- Big Data, then AI/ML systems
- New “cloud native” DBs
→ Store and process business or operational data for insights

2020 +: Timeliness
- The world’s real-time ops moving to the cloud; e.g., retail, financial trading, gaming, two-sided markets, eSports, AVs, ...
- New constraints are following; e.g., deadlines, event ordering, ”temporal fairness”, ...
→ Need tech where “time” and “time-sensitivity” are a P0 feature
Our Approach: Time Perimeters

Synchronize clocks at just the desired nodes (the “perimeter”)
  • No need to sync all intermediate clocks → enables scaling in size and distance

Timestamp packets as they pass through the perimeter
  • This could be in either direction
  • Use the timestamps to make scheduling decisions at the perimeter or other nodes
Delivering Timeliness Using Time Perimeters

Time perimeters enable powerful solutions in Distributed Systems and Networking
• Event ordering/scheduling (e.g., databases, distributed ledgers, snapshotting) → Lamport’s total ordering of events at different nodes can be solved (up to clock fuzz)
• Building deterministic and jitter-free networks (e.g., CloudEx) → Resequencing and Hold/Release buffers can achieve this
• Large-scale monitoring and control (e.g., SIMON, On-Ramp)
• Take an action in a precise time window (e.g., sell stock for $X only in a specific time window)

Two major enhancements to scheduling enabled by Time Perimeters
• Scheduling decisions can be made based on absolute values as opposed to relative values
  • In Networking and Systems, scheduling decisions are based on comparisons (shortest, longest, oldest, etc)
• Scheduling decisions can be based on non-local information (e.g., timestamp taken elsewhere)
  • Typically, scheduling decisions are based on local state variables, not global variables
This Session

**CloudEx: A teaching and research tool, Vig Sachidanda and Jinkun Geng**
- Experiences and enhancements from using it in CS 349F in Fall 2020

**On-Ramp: Managing congestion from the network’s edge, Shiyu Liu**
- Experience from a trial at Facebook

**Storage stories: Manoj Wadekar, Facebook**
- The evolution of storage in large-scale DCs and its implications