NTP Synchronization Algorithm

Distributed Systems - Class Presentation

Aditya Morolia, Kalp Shah, Vivek Puar

January 8, 2021

Clock Synchronization

Process of ensuring that physically distributed processors (systems) have the same notion of time.

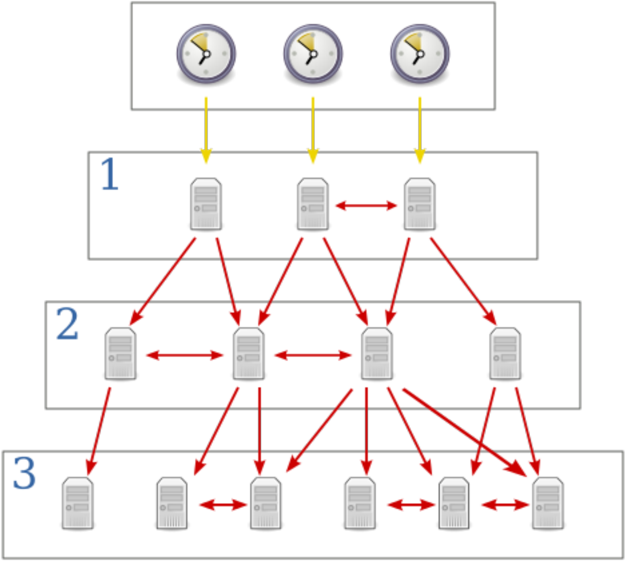
Need for clock synchronisation

- We need timestamps or intervals and orderings of time between events for processing.
- Example: Updates should be before accesses (ordering).
- Replace communication with local computation.
- Implementing timeouts in protocols or algorithms.

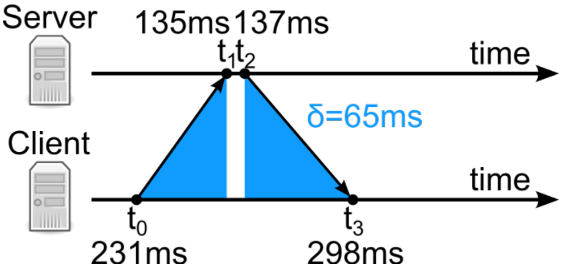
Network Time Protocol – Highlights

- NTP later structure: Multiple strata starting at 0/1 at the original time sourse. Each node syncs with it's parent node.
- Benefits of NTP: Large adjustments made quickly and then smaller adjustments keep happening over time.
- insane time (part of protocol/implementation)
- Uses UDP (user datagram protocol). Hence low latency.

Network Time Protocol – Schematic



Interaction



Protocol

Figure 3.9 Offset and delay estimation [15].

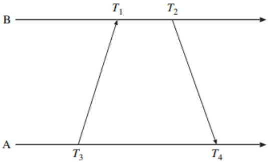
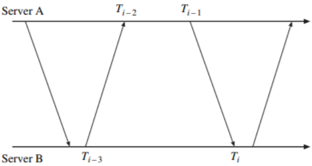


Figure 3.10 Timing diagram for the two servers [15].



Let $a = T_1 - T_3; b = T_2 - T_4$. Then,
Clock offset $\theta = (a + b)/2$
Round trip delay $\delta = a - b$

Algorithm

- A pair of servers in symmetric mode exchange pairs of timing messages.
- A store of data is then built up about the relationship between the two servers (pairs of offset and delay).
- Specifically, assume that each peer maintains pairs (O_i, D_i), where O_i is the offset θ and D_i is the transmission delay of the two messages δ
- The offset corresponding to the minimum delay is chosen. Specifically, the delay and offset are calculated as follows. Assume that message m takes time t to transfer and m' takes t' to transfer.

Algorithm ... continued

- The offset between A's clock and B's clock is O. If A and B's local clock times are $A(t)$ and $B(t)$, we have

$$A(t) = B(t) + O$$

$$T_{i-2} = T_{i-3} + t + O$$

$$T_i = T_{i-1} - O + t'$$

Assuming $t = t'$, we can estimate offset

$$O_i = (T_{i-2} - T_{i-3} + T_{i-1} - T_i)/2$$

And Round trip delay

$$D_i = (T_i - T_{i-3}) - (T_{i-1} - T_{i-2})$$

- Retain 8 latest pairs of (O_i, D_i) and choose the offset corresponding to the minimum round trip delay to estimate O.

Thank You!