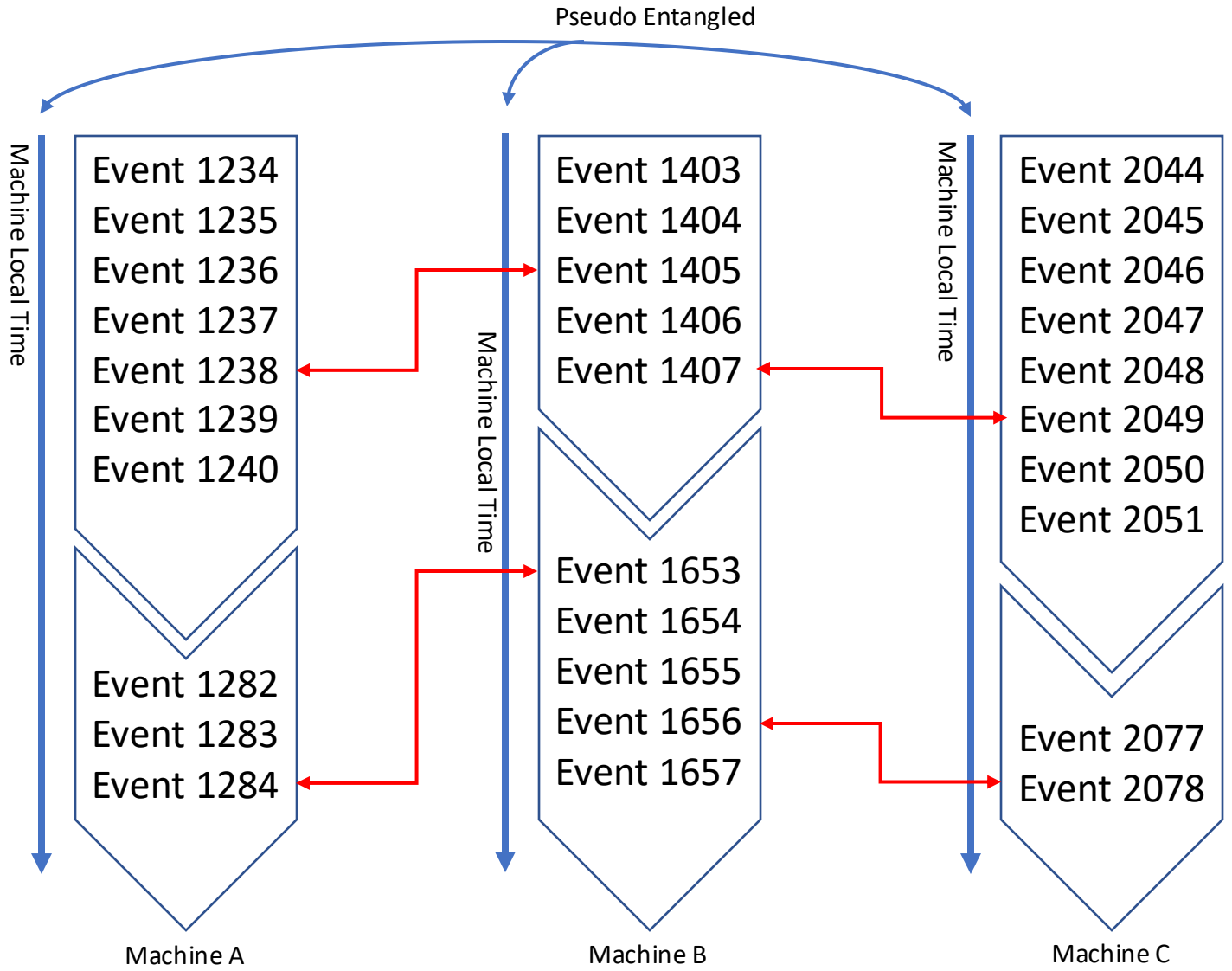
Tangle

Associating Events Across Machines

Problem Formulation

- Each machines runs on its own local time scale
- Events on each machine are time stamped with the local time scale
- It is necessary to associate event time stamps between machines
- Universal Time Coordination (UTC) can be used as a common time scale on all machines
- It is necessary to convert local time scale to UTC on each machines with the necessary precision
- Necessary precision is driven from the fastest event logging frequency
- For multiple machines, the necessary precision increases with the scale
- It is important to calculate the window of uncertainty on each machine

Correlate Events Between Machines



Precision Requirement

- CPU level
- OS (Kernel) level
- Distributed System level

Precision Requirement at CPU level

- Nyquist sampling theorem
  - Sampling interval required to avoid aliasing
  - Sampling frequency should be at least twice the highest frequency contained in the signal
- Frequency in event occurrence
  - Instruction Latency
  - Instruction Throughput

// mov = 1 CPU cycle

// xchg = 3 CPU cycles

// rdtsc = 1 CPU cycle

A CPU with a clock speed of 3.2 GHz executes 3.2 billion cycles per second
That is a period of about 310ps

Precision Requirement at OS level

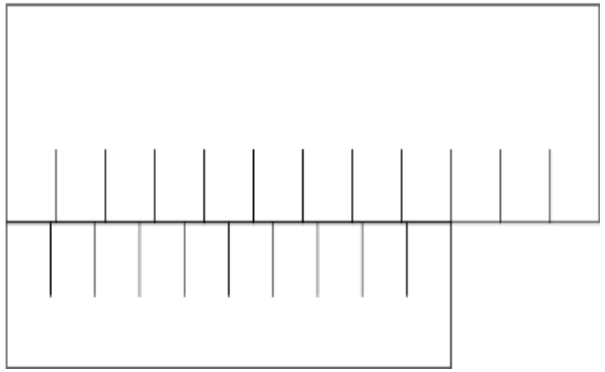
- dmesg

```
[52603.373642] {38}[Hardware Error]: event severity: corrected
[52603.373643] {38}[Hardware Error]:  Error 0, type: corrected
[52603.373644] {38}[Hardware Error]:   section_type: PCIe error
[52603.373644] {38}[Hardware Error]:   port_type: 4, root port
[52603.373645] {38}[Hardware Error]:   version: 3.0
[52603.373645] {38}[Hardware Error]:   command: 0x0547, status: 0x0010
[52603.373646] {38}[Hardware Error]:   device_id: 0000:b7:01.0
[52603.373647] {38}[Hardware Error]:   slot: 255
[52603.373648] {38}[Hardware Error]:   secondary_bus: 0xb8
[52603.373648] {38}[Hardware Error]:   vendor_id: 0x8086, device_id: 0x352a
[52603.373649] {38}[Hardware Error]:   class_code: 060400
[52603.373649] {38}[Hardware Error]:   bridge: secondary_status: 0x0000, control: 0x0013
```

System Logging is based on clock_boottime (clock_minotone_RAW) with a quanta on 1us
Events occur faster than the quanta of 1us (aliasing)

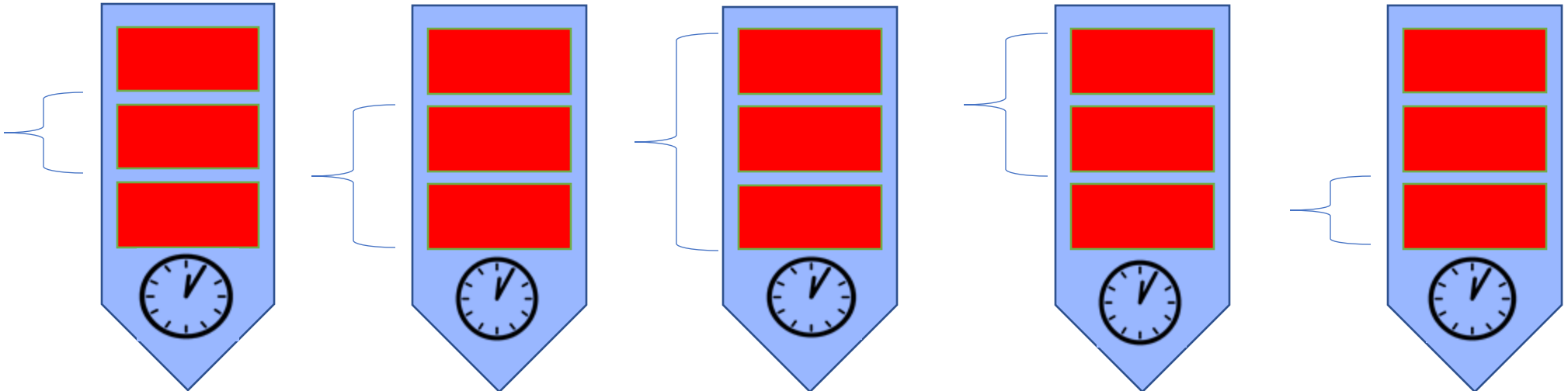
Challenges and the Precision Requirement

- Vernier acuity

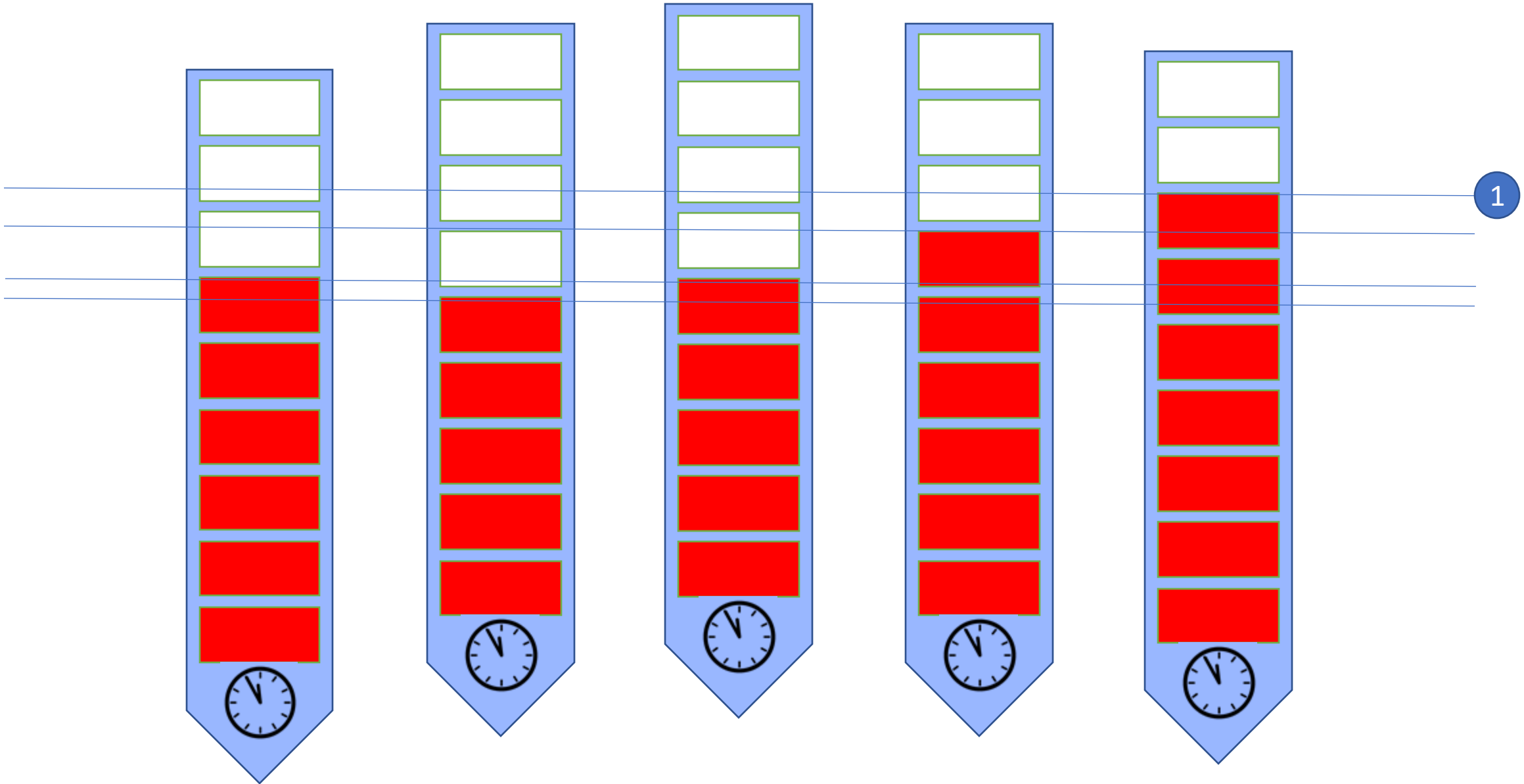


- Compounding of Events

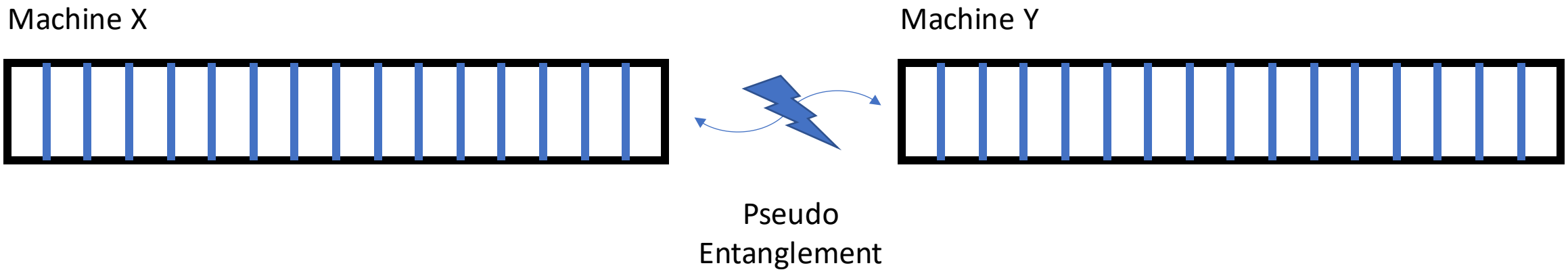
$$1^n = 1$$



Use case for correlated event logging



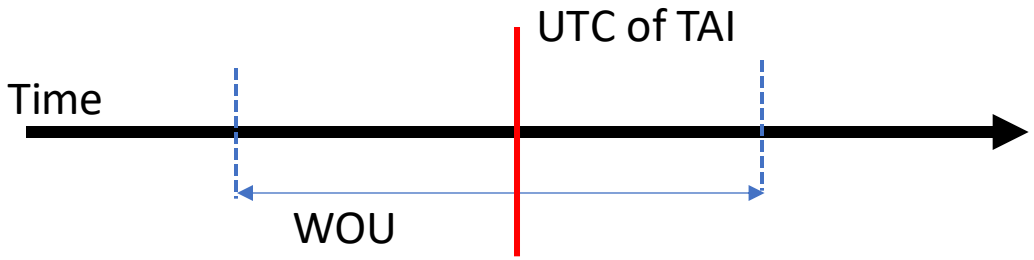
What comes out of Precision Time Sync?



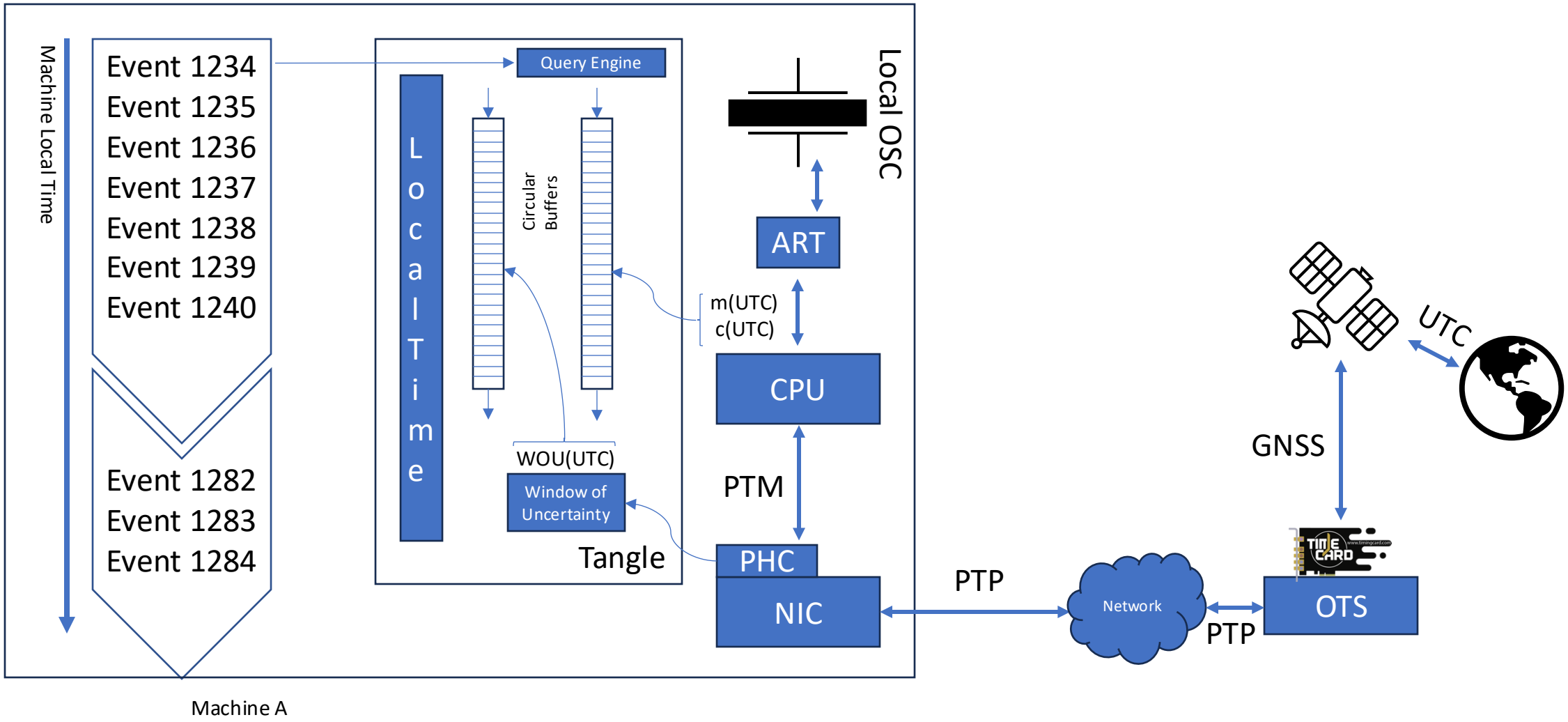
Machine X and Y are any two machines across the globe or inside a local network

Pseudo Entanglement: Probabilistic Entanglement of two Registers (Machine Y and Y) within the Windows of Uncertainty

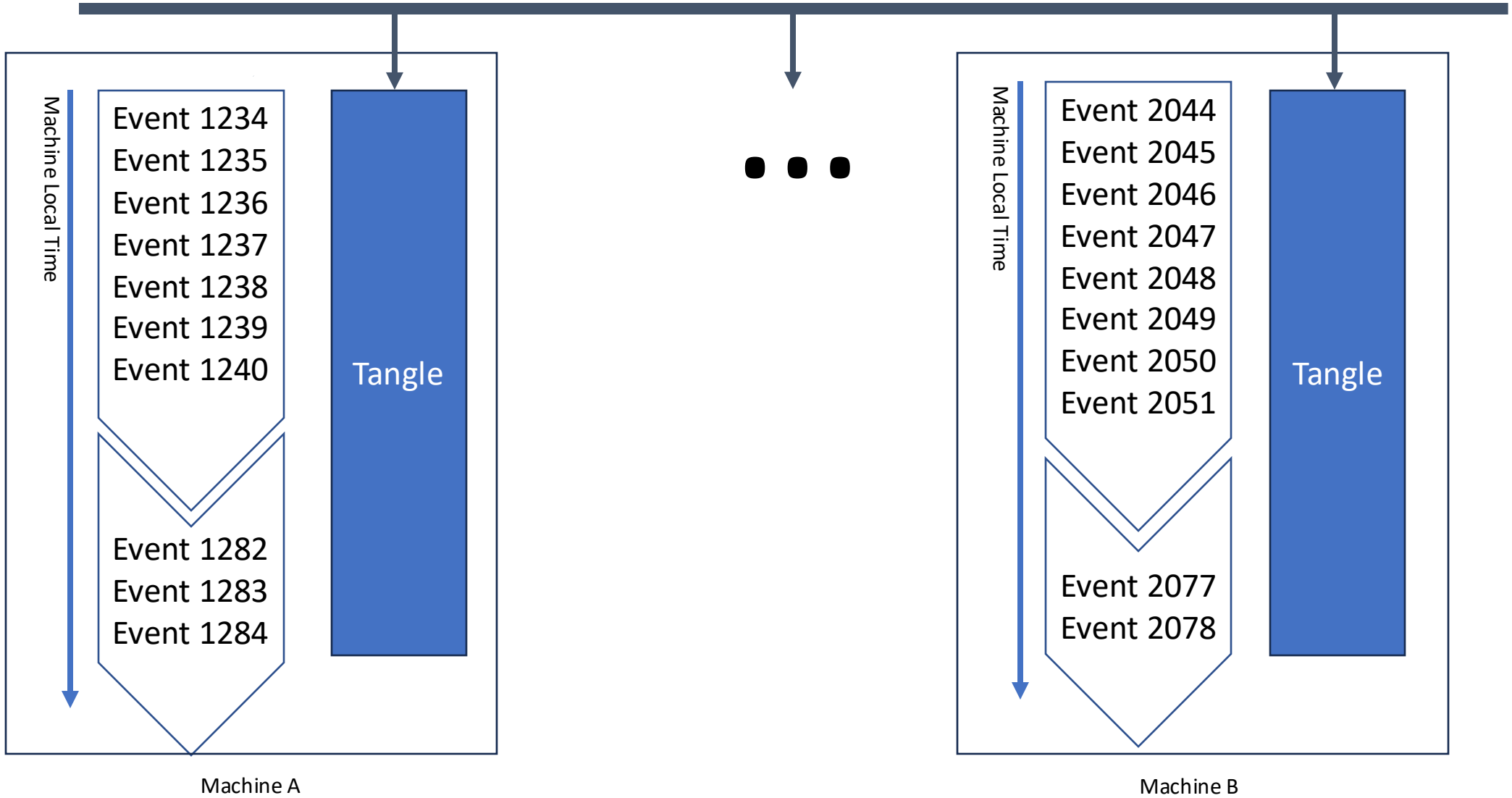
Window of Uncertainty: An ongoing estimation of a time interval that UTC (or TAI) sits inside it (with a given probability)



Architecture



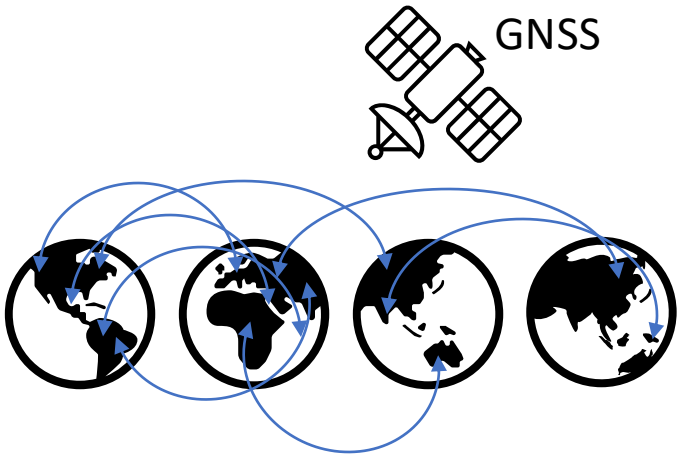
Multiple Systems

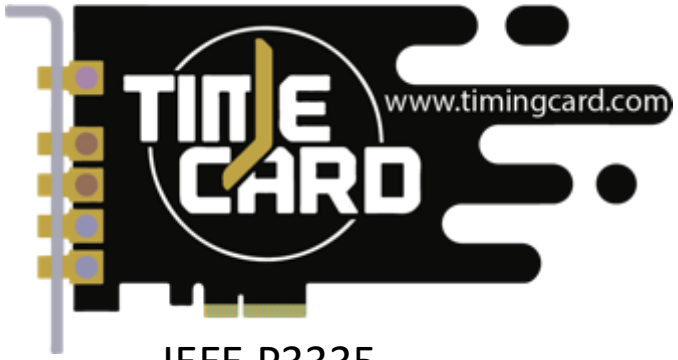


Functions

- Identify concurrent event in another machine[s]
- Find the timestamp of an event in another machine[s]
- Chronologically Rank a given event across machines
- Measure the one-way-latency between machines
- Identify concurrent events with one-way-latency consideration
- Trace chronological order for sequence of events
- Benchmark machines by precise runtime measurement
- Directly utilize RDTSC for maximum precision in event timestamping

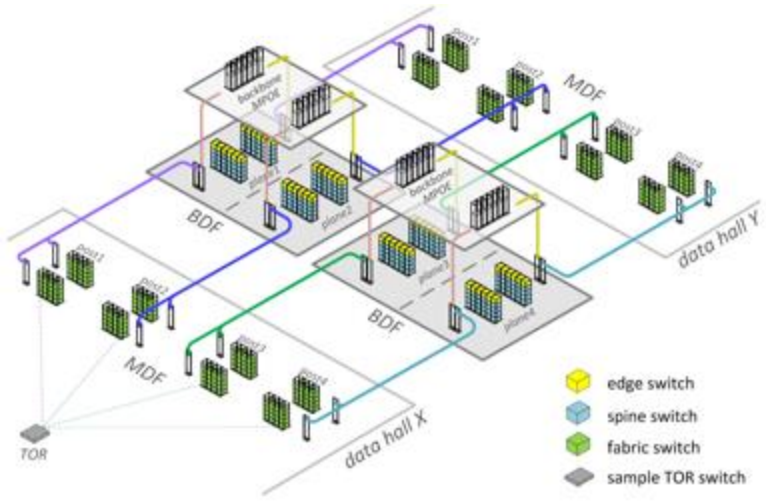
Precision Time Sync Across Different Domains





IEEE P3335

Global



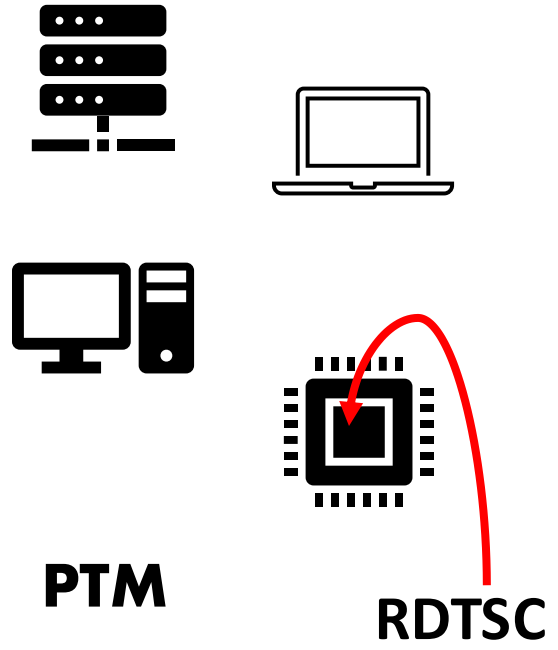
PTP

Data Center IEEE 1588

Local Network

RDTSC — Read Time-Stamp Counter

Opcode*	Instruction	Op/En	64-Bit Mode	Compat/Leg Mode	Description
0F 31	RDTSC	ZO	Valid	Valid	Read time-stamp counter into EDX:EAX.



Inside Machines

Why is PTP+PTM needed?

Applications for Precision Time Synchronization in Data Centers

- Distributed Databases
- Distributed Load Balancers
- In Network Telemetry
- Distributed AI Systems

Time Precision and Applications