A Quest Against Time

- Why timekeeping is hard
- What we can do without guest help
- What we can do with guest help

PART 1 – TIME IS HARD

PART 1 – TIME IS HARD

- Not this hard...

$$\Delta p_2(NT_s) = \sum_{i=1}^{N}[k_1 e^{-(N-i)T_s/T}(1 - e^{-(T_s/T)})\Delta p_1(iT_s)] +$$

$$\sum_{i=1}^{N}[-k_2 e^{-(N-i)T_s/T}(1-e^{-(T_s/T)})\Delta M_2(iT_s)] - \left[\frac{-k_2 T_2}{T}(1-e^{-(T_s/T)}) \right.$$

$$\left. [\sum_{i=1}^{N}\Delta M_2(iT_s)e^{-(N-i)T_s/T}] + \frac{k_2 T_2}{T}\Delta M_2(NT_s)\right] \quad (25)$$

$$\Delta M_1(NT_s) = \sum_{i=1}^{N}[e^{-(N-i)T_s/T}(1 - e^{-T_s/T})\Delta M_2(iT_s)] +$$

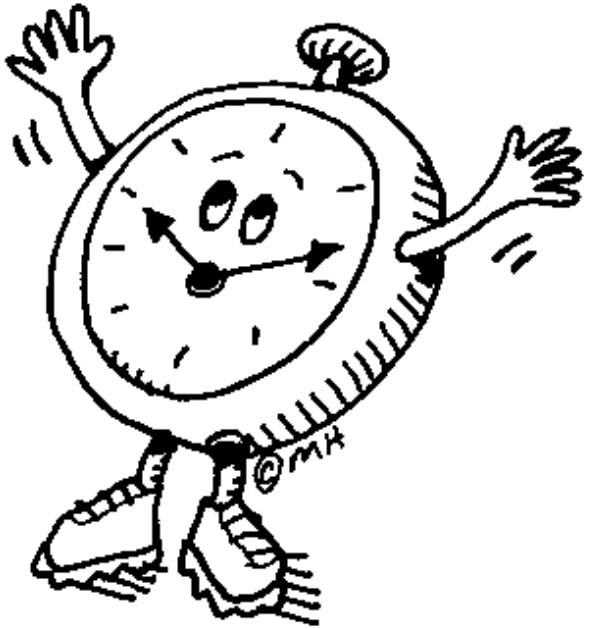
$$\left[-\frac{T_1}{T}(1 - e^{-T_s/T})[\sum_{i=1}^{N}\Delta p_1(iT_s)e^{-(N-i)T_s/T}]\right] + \frac{T_1}{T}\Delta p_1(NT_s)$$

$$(26)$$

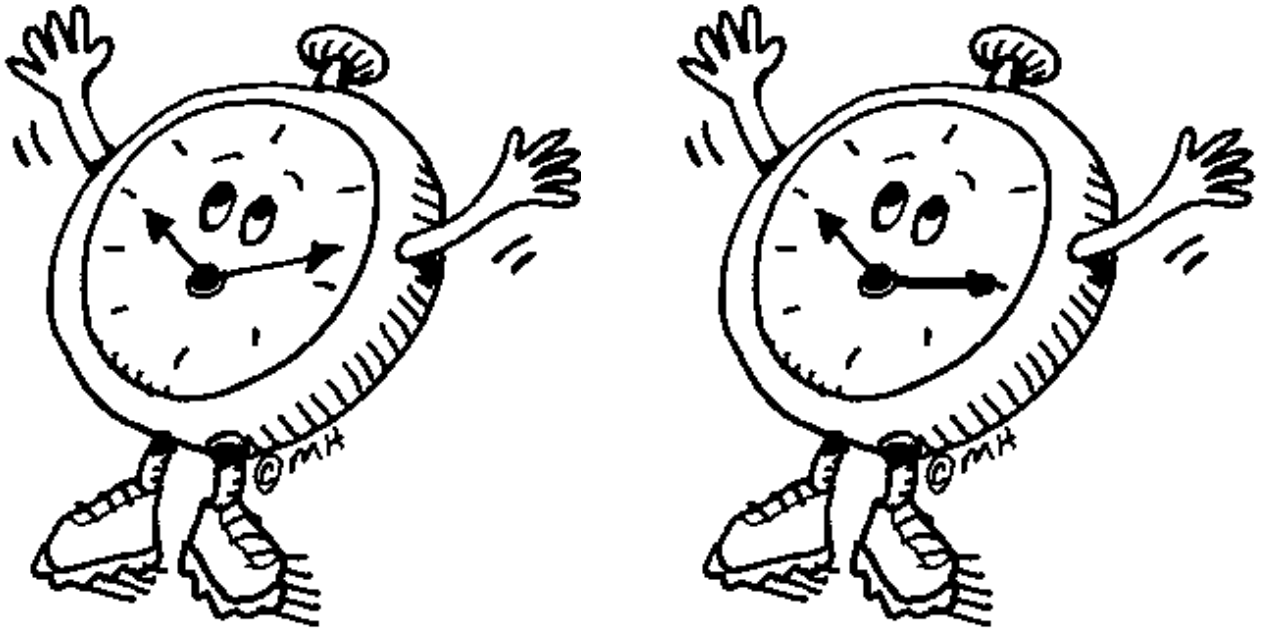
PART 1 – TIME IS HARD

- Not this hard...
- It's worse

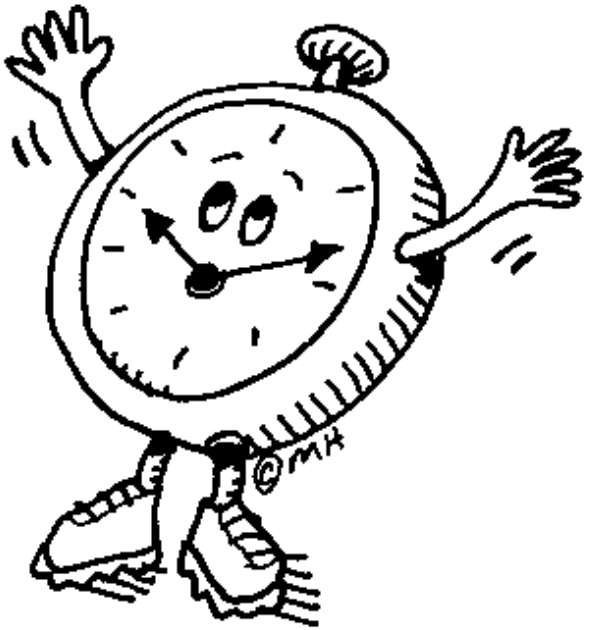
Every measurement is an observation...

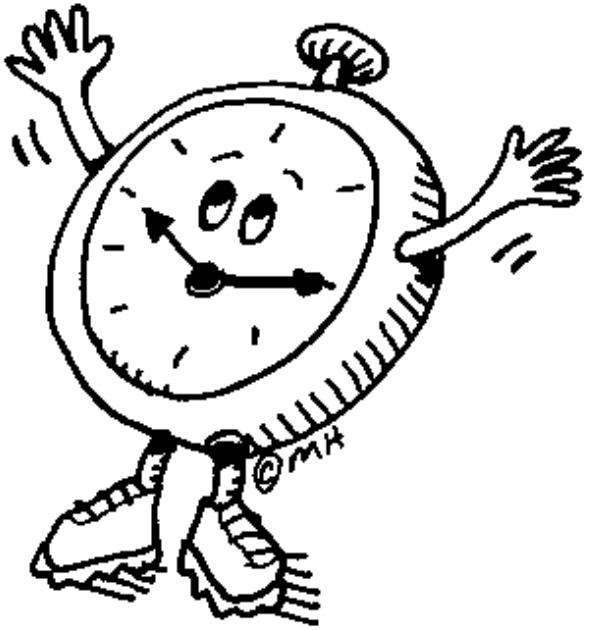


And every observation must be consistent....



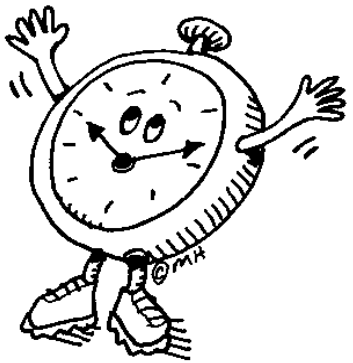
Not just with itself, but with other clock interrupts

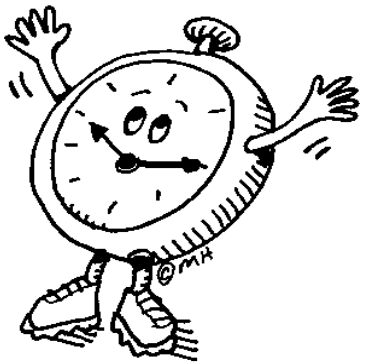




PIT LINE

And there are many of these





PIT LINE

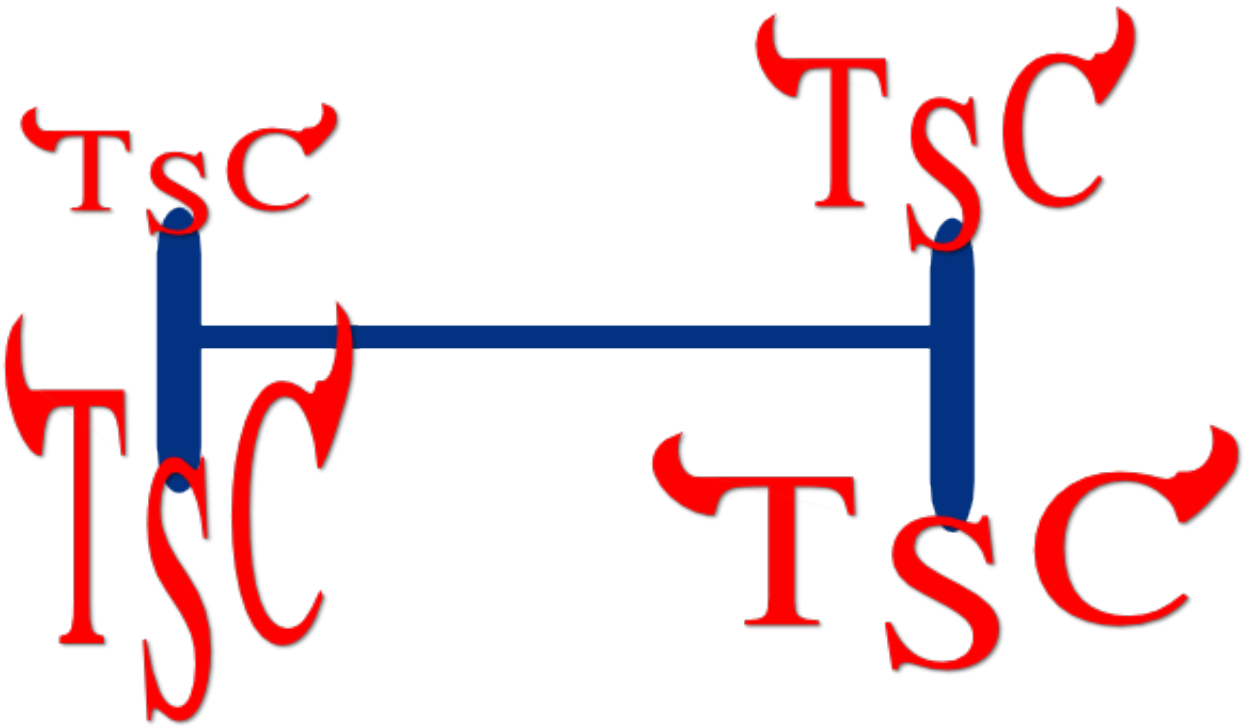
HPET LINE

APIC LINE

Some are local entities

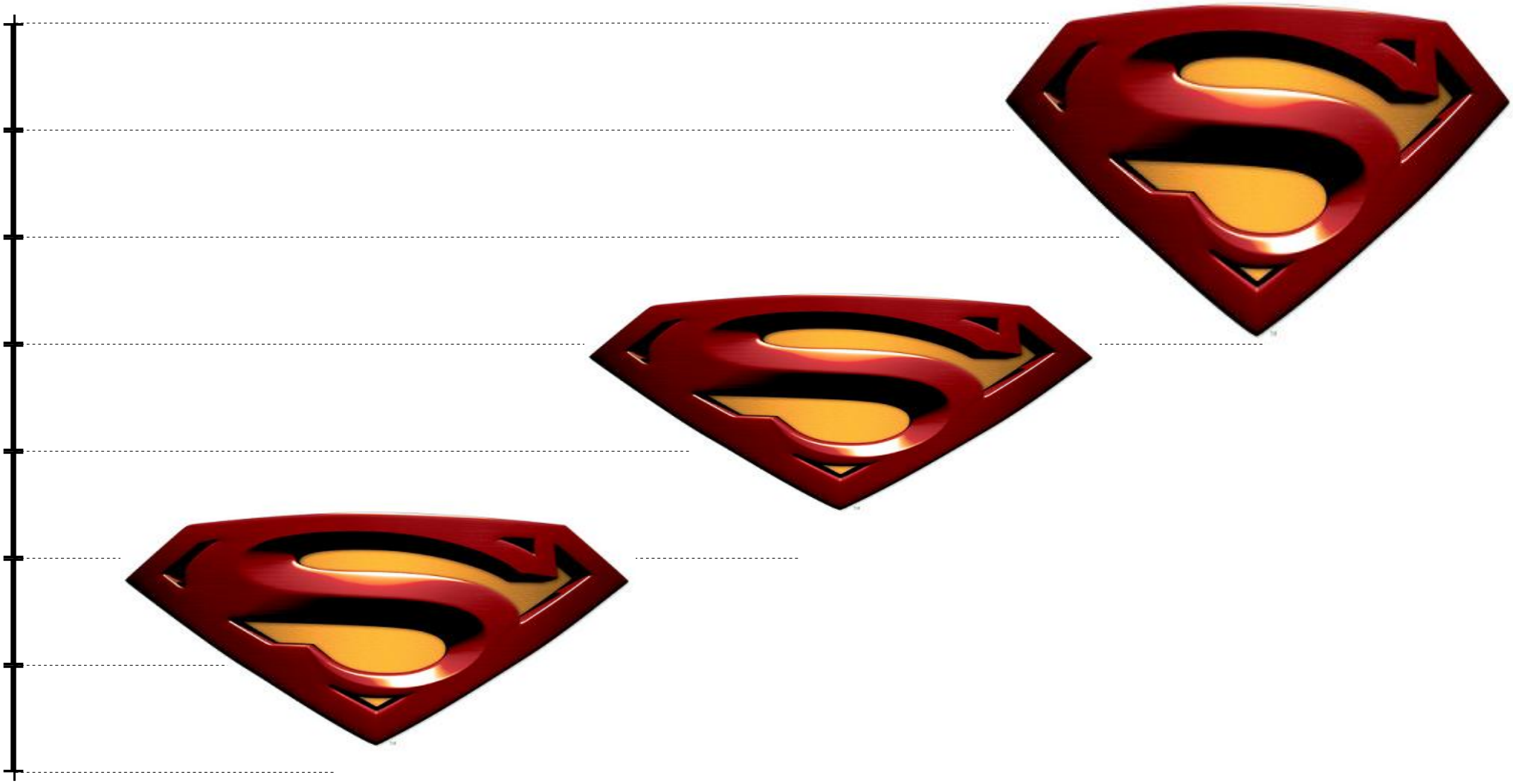
And reaching agreement is hard (inter-cpu drift)

And reaching agreement is hard (inter-socket drift)



And reaching agreement is hard (thermal effects)

And reaching agreement is hard (super-scalar execution)



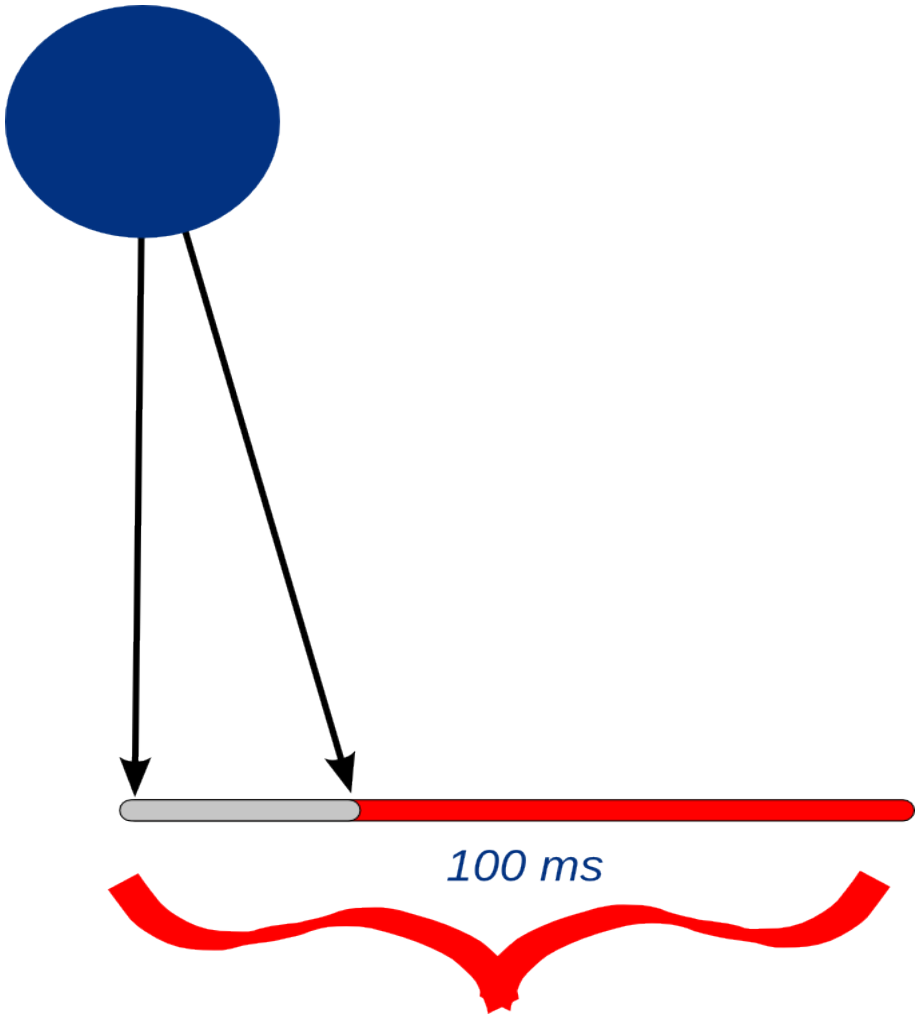

It is hard on baremetal too

On virt, assumptions break

PART 2 – On our own

Interrupts delivered, guest is out

But it still believe it made it

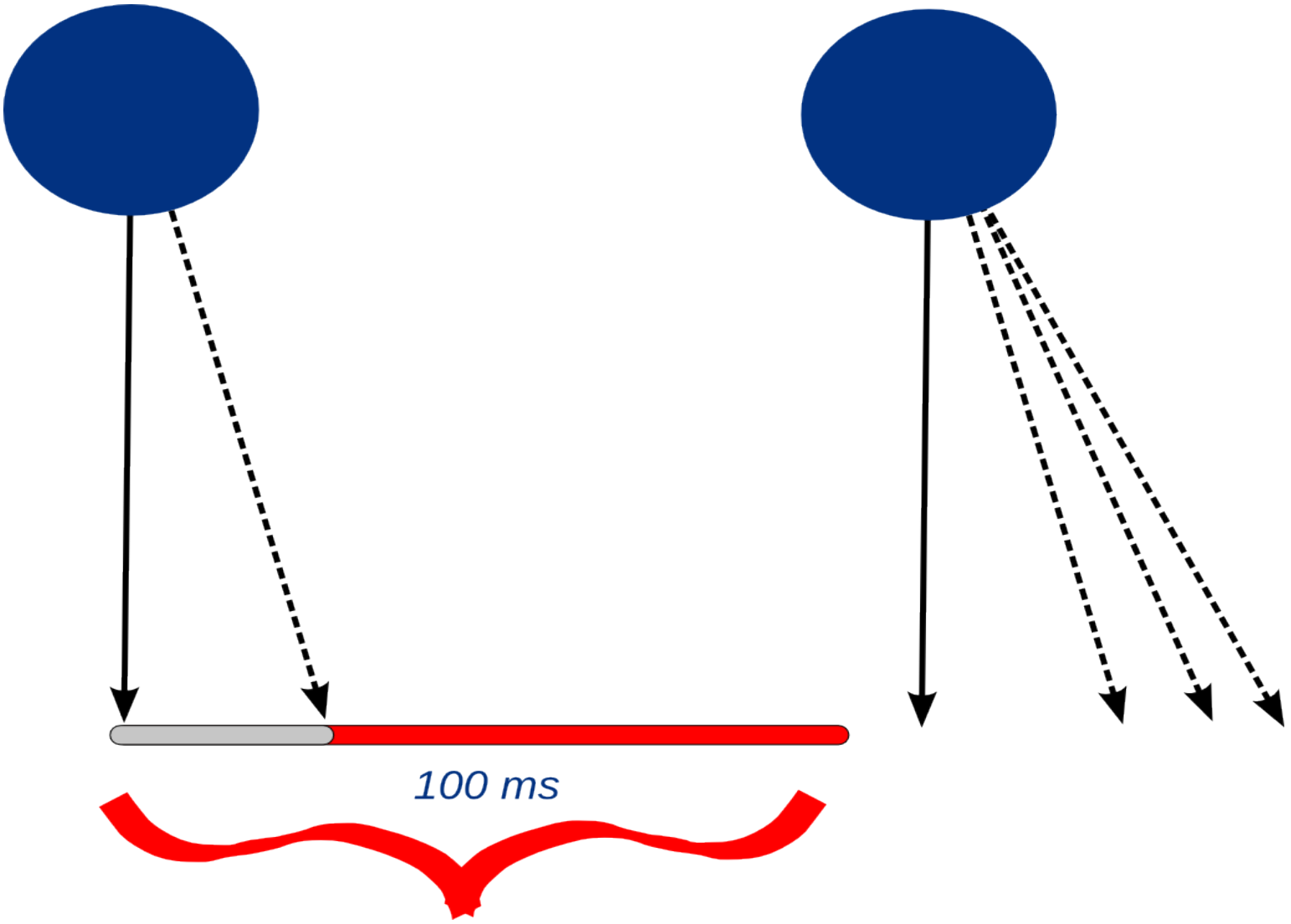


When to deliver next interrupt, hard target

When did guest really process it?

When did guest really process it?

Next time, send many



Takes a lot of cpu

Part 3 – Guest cooperation

Ideally, not rely on interrupts

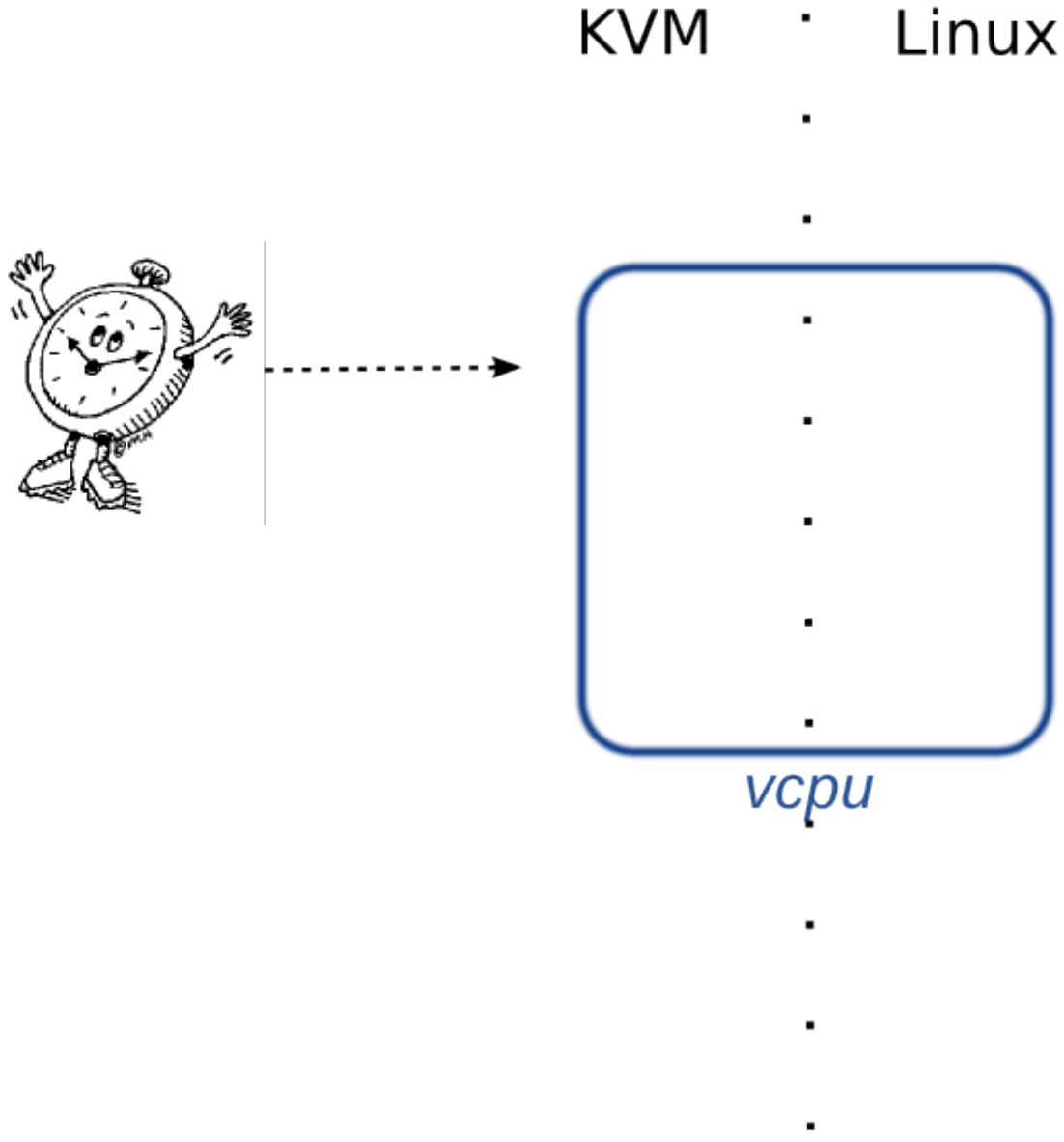
- Read clock timestamp directly (modern linux clocksources)

But if we might, better to compensate in the guest

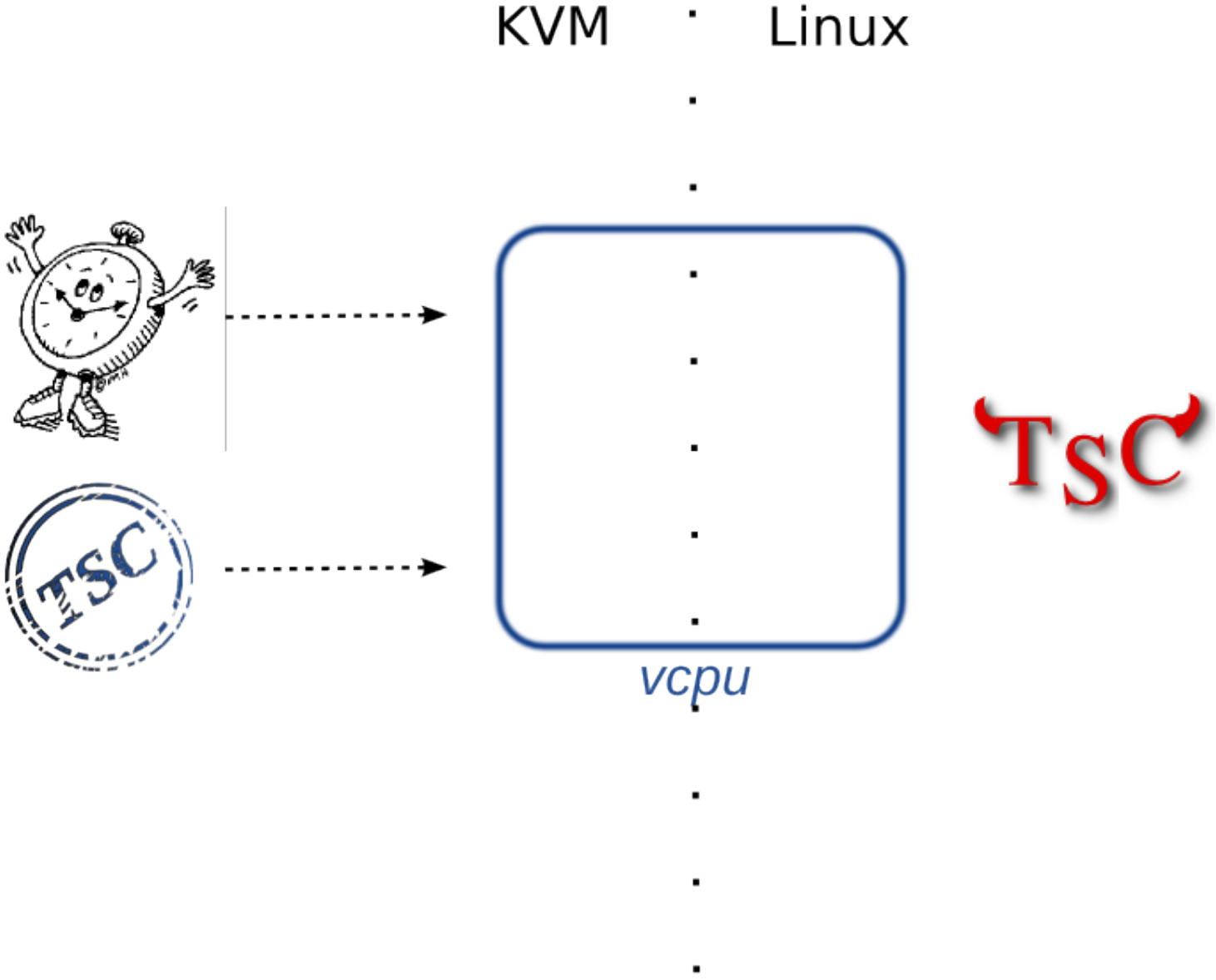
- Read clock timestamp directly (modern linux clocksources) => and then figure out how many ticks we should account.


KVM

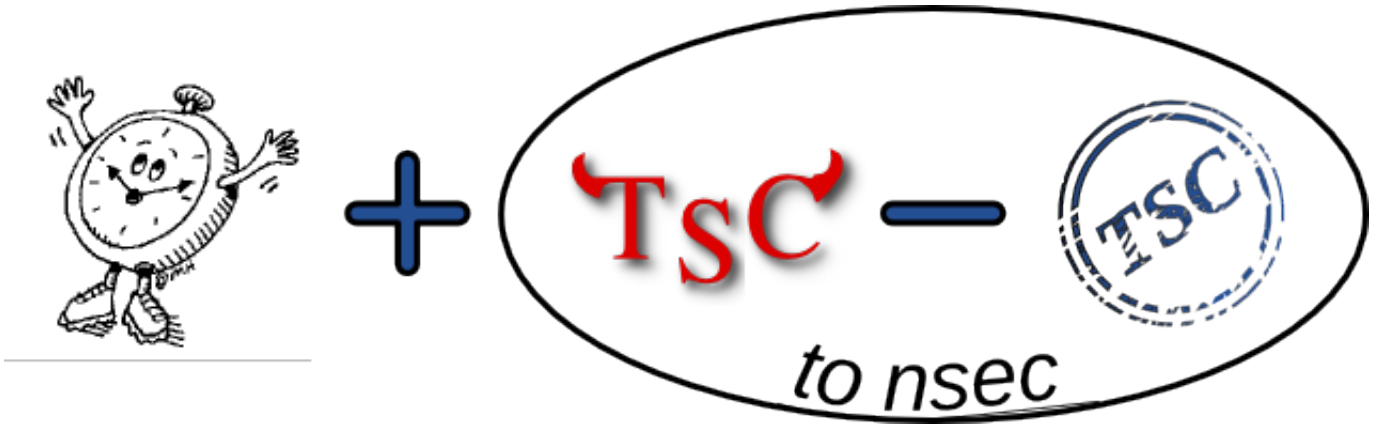
Hypervisor tells time



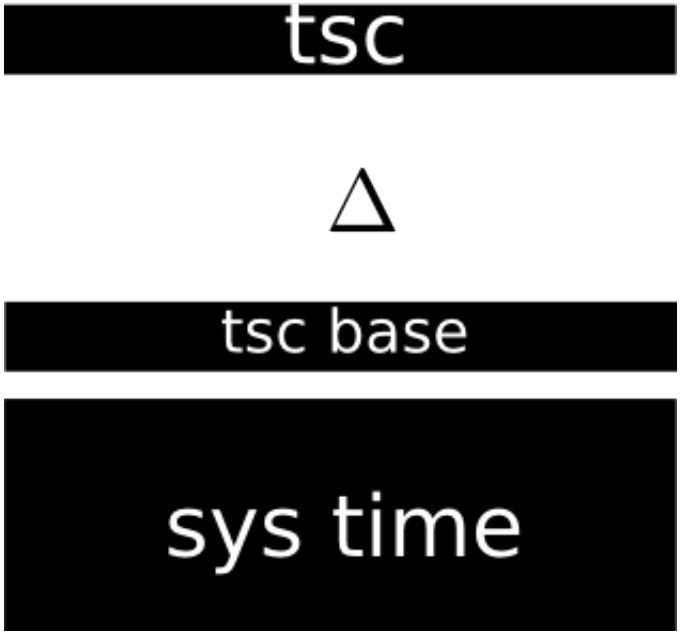
Adjust locally with tsc



Adjust locally with tsc



The picture

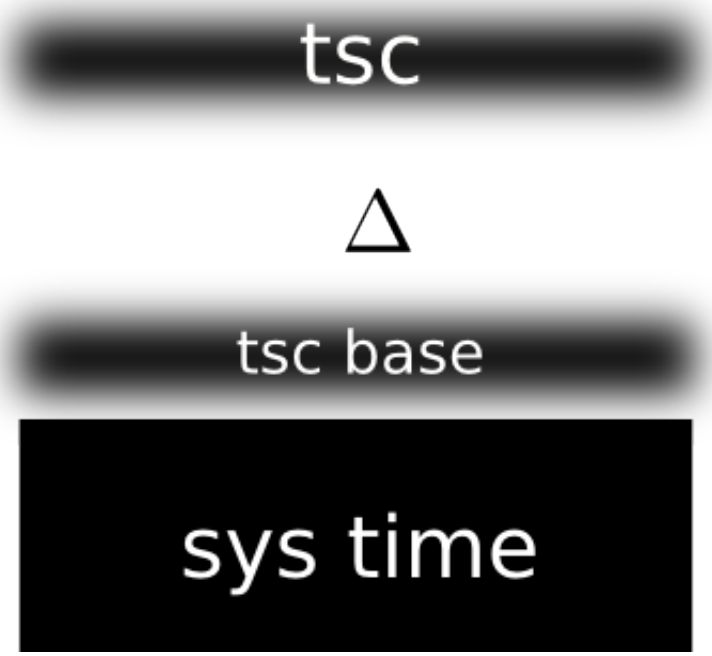


Must be done carefully

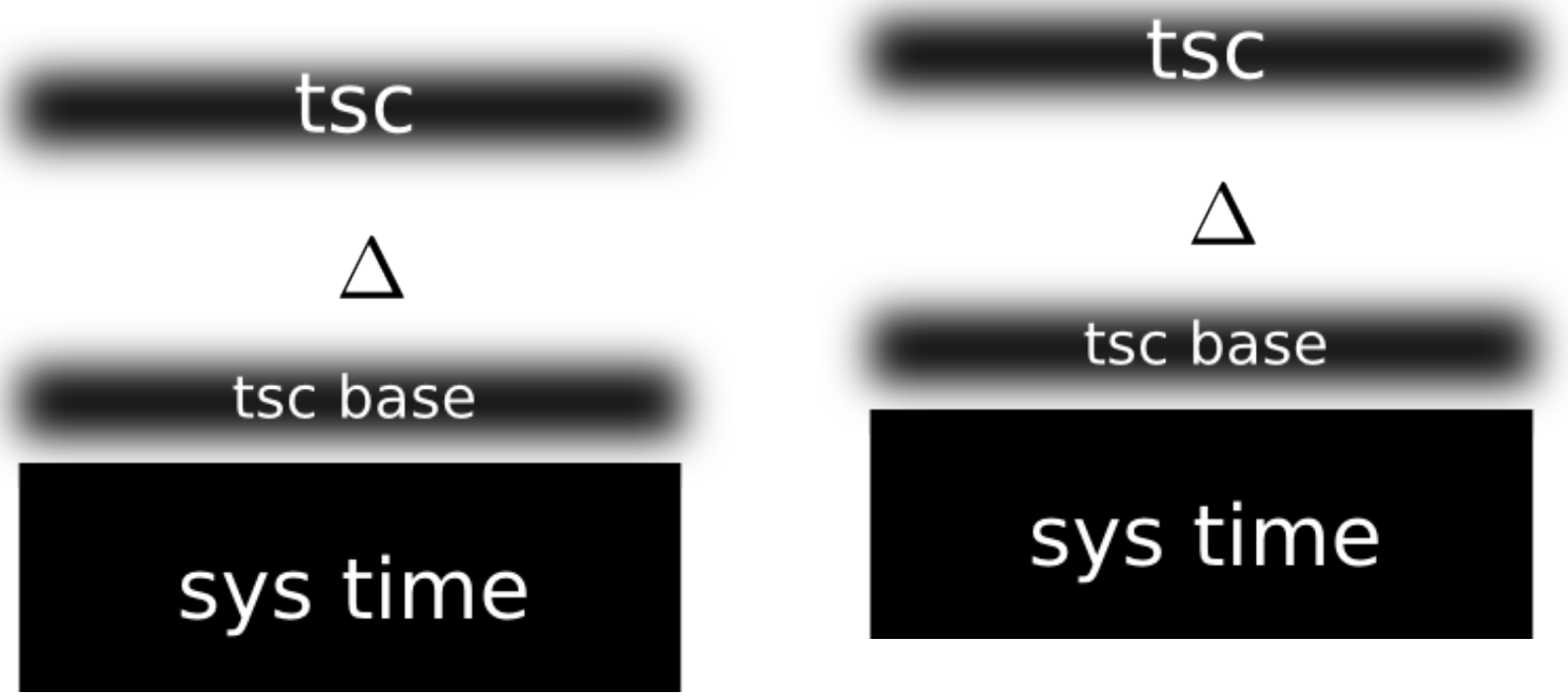
tsc and host clock may run at different resolutions, usually faster

tsc has issues

Even if everything works ok



Recalibration has serous issues, same as SMP



Worst case? Hit it with a hammer

Thank you