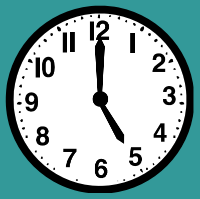


Timing over packet networks – real solutions to real problems

February 2010

Presented by:

Yaakov Stein

Chief Scientist

RAD
data communications

What is this talk about ?

About 30 minutes …

but how do we know how much time 30 minutes is ?

Disclaimer

TIMING is a complex (i.e. highly mathematical) subject

Von Neumann used to say :

a technical lecture should optimally be 1 μCentury in length
(1 millionth of 100 years ≈ 52 ½ minutes)

Guy Kawasaki says :

an optimal marketing pitch obeys the **10-20-30** rule
10 slides / **20** minutes / **30** point fonts !

We'll take the middle ground

30 minutes but too many slides and too small fonts

Timing types

When we say ***timing***

we usually mean (at least) one of three different things

1. **Frequency**
2. **Uncalibrated time** (often *ambiguously* called phase)
3. **Time of Day** (ToD)

Each of these

- fulfills a distinct **need** (application)
- requires *additional* **means** to obtain (distribute)

Basic definition – the **second**

The second was once defined to be 1 / 86,400 of an *average* day

Unfortunately, the earth's rotation rate varies during the year

So in 1956 the second was redefined to be 1 / 315,569,259,747 of a year (specifically, the year 1900)

This is the basis of **UT1** (Universal Time)

Unfortunately, the earth's rotation rate varies from year to year

So in 1967 the second was redefined to be 9,192,631,770 cycles of the radiation emitted by the transition between the two hyperfine levels of the ground state of the Cs-133 atom

In order to maintain accuracy the measurements of many labs are combined to form **TAI** (International Atomic Time)

UTC (Coordinated Universal Time) is a compromise

It differs from TAI by an integral number of seconds
but is kept within 0.9 seconds of UT1
by introducing *leap seconds*

Basic definition – **frequency**

Frequency is defined to be the number of times a periodic phenomena repeats in one second

It is expressed in inverse seconds = **Hertz** (Hz)

- AC current is supplied at 50 or 60 Hz
- we hear sounds up to about 20 or 25 KHz
- AM broadcast radio is transmitted between 0.52 and 1.61 MHz
- WiFi, bluetooth, and μwave ovens operate at about 2.4 GHz

Deviation of a frequency from its nominal value is expressed as a FFO (Fractional Frequency Offset) in

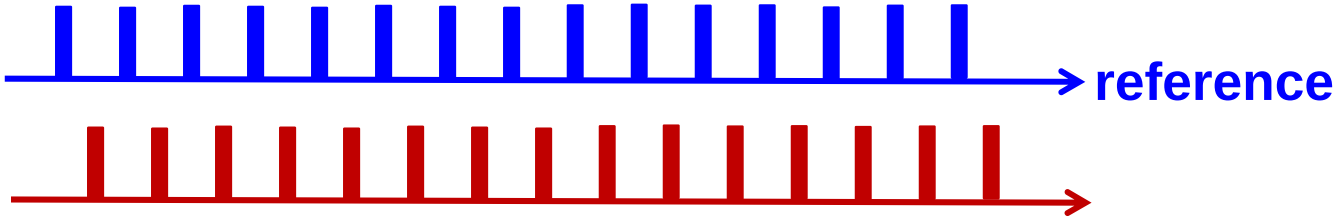
- parts per million (**ppm**) or
- parts per billion (**ppb**) or
- 10^{-n}

Basic definition – **phase lock**

To ensure that a periodic phenomenon has the same frequency as another phenomenon (the *reference* frequency)

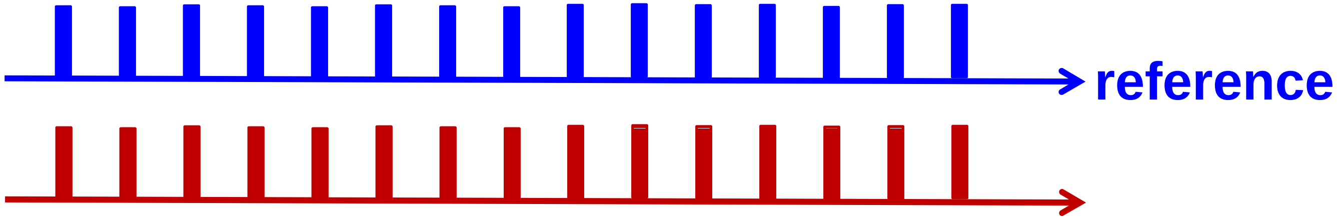
It is enough to ensure that an event occurs for every reference event

This is called ***frequency lock***



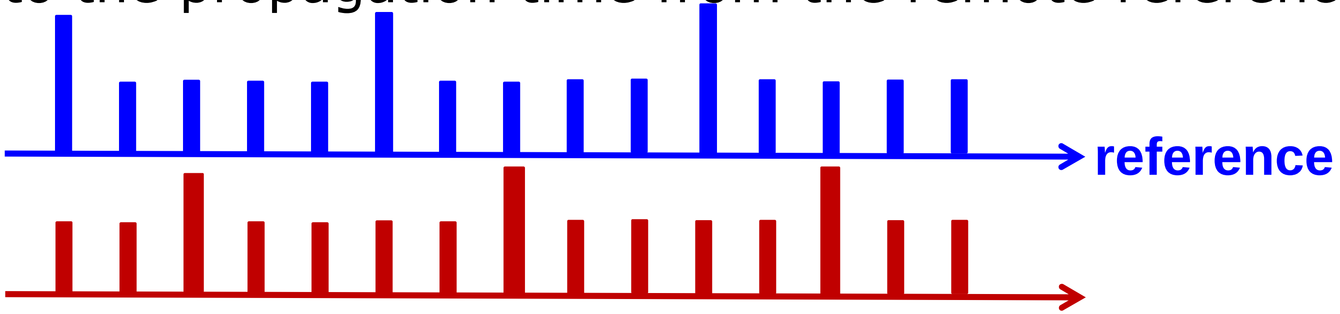
A stronger condition is ***phase lock***

where the events occur at exactly the same times

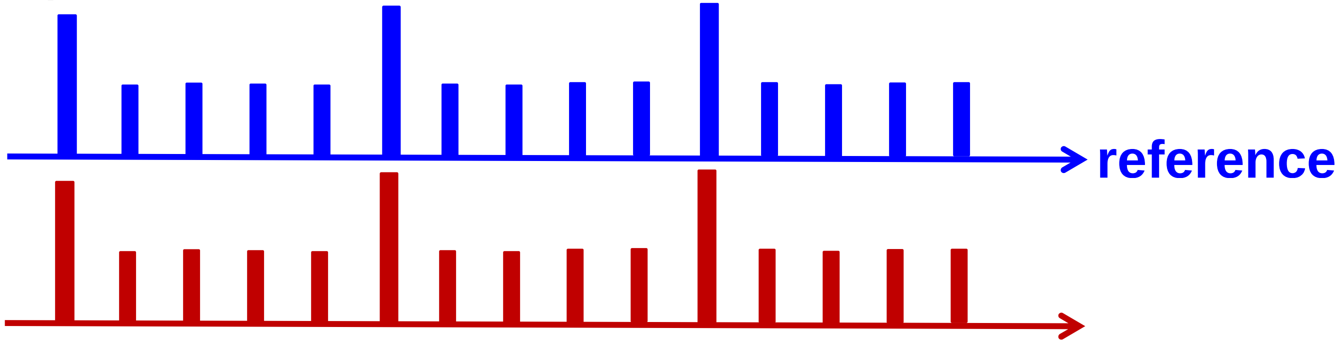


Basic definition – **uncalibrated time**

When a phenomena is phase locked to a *remote* reference
the "beginning of second" markers may not be aligned
due to the propagation time from the remote reference signal



When they *are* (although we still don't know which second is starting)
we say that we have locked **uncalibrated time**



and we can output a **1pps** uncalibrated time signal

Basic definition – **Time of Day**

Once, each country defined a local time of day (hh:mm:ss)

In 1884 **GMT** was introduced as a world standard
and the world was divided into time-zones

In 1972 GMT was replaced by **UTC**, and each country :

- decides on offset from UTC (usually full hours)
- decides when/whether to use daylight savings time (summer time)

Unfortunately, several organizations claim to dictate UTC

The two most important ones are :

- NIST (Boulder, Colorado) – used for GPS system
- UTC Observatoire de Paris – used by the Galileo system

The difference between these two is usually < 20 nanoseconds

The **needs** **(non-exhaustive list)**

Frequency is needed for applications that need to :

- recover periodic phenomena (e.g. bit streams)
- transmit with spectral compatibility
- accurately measure :
  - time durations (differences)
  - periodicities
  - distances (using the constant speed of light)
- perform actions at a constant rate

Uncalibrated time is needed for applications that need to :

- perform an action *just in time*
- transmit in bursts w/o interfering with others
- tightly coordinate execution with multiple neighbors
- triangulate to find location

Time of Day is needed for applications that need to :

- precisely schedule events
- timestamp events
- prove an event tool place before/after another event

Example applications

Frequency

- synchronous (TDM) networking – bit recovery
- delivery of frequency to lock RF of GSM base-stations
- calibration (police radars, parking meters, …) and metrology
- FDD, FDMA arbitration

Uncalibrated time

- factory equipment coordination
- TDD, PON, TDMA arbitration
- optimization of networking resources/paths
- transmit in bursts w/o interfering with others
- low power sensor networks
- GPS

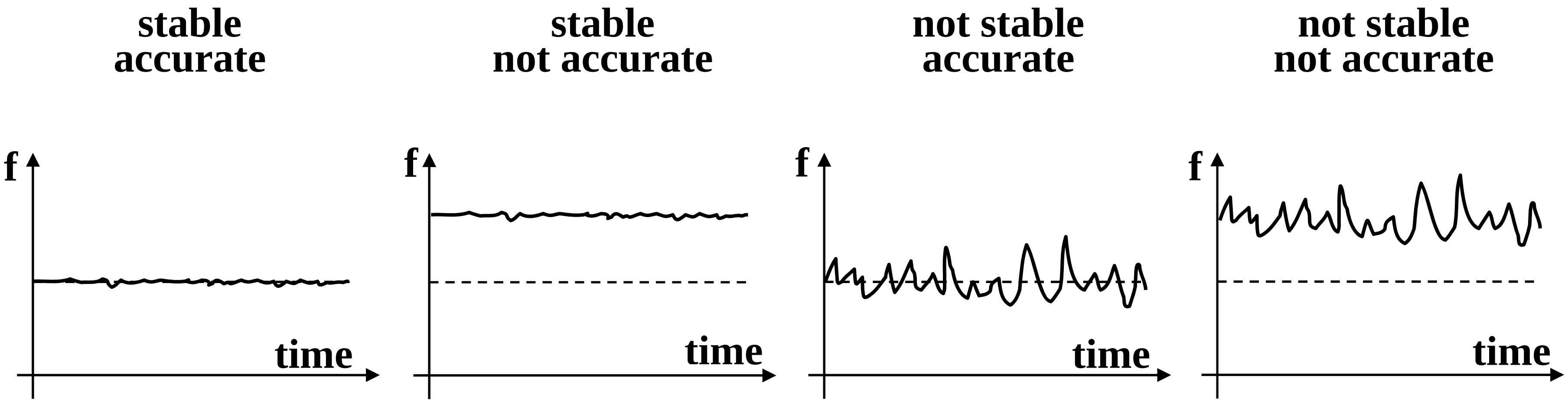
Time of Day

- delivery of time to set clocks
- time-stamping financial transactions
- smart grids (time-based metering)
- legal uses of ToD

Frequency - stability and accuracy

The performance of a system may depend on its frequency

stability and/or **accuracy**



Frequency distribution **jitter**

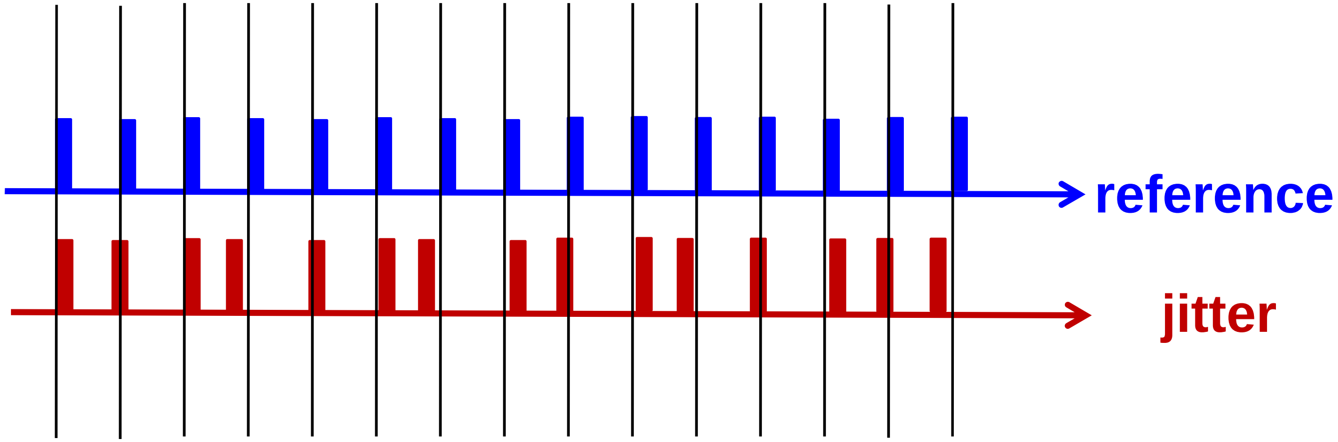
We need metrics that enable accurate performance predictions

- FFO is too blunt a tool for highly accurate frequency distribution

It is conventional to distinguish between **jitter** and **wander**

We start by measuring **TIE** (Time Interval Error)

Jitter is event-event (high frequency) fluctuations in TIE

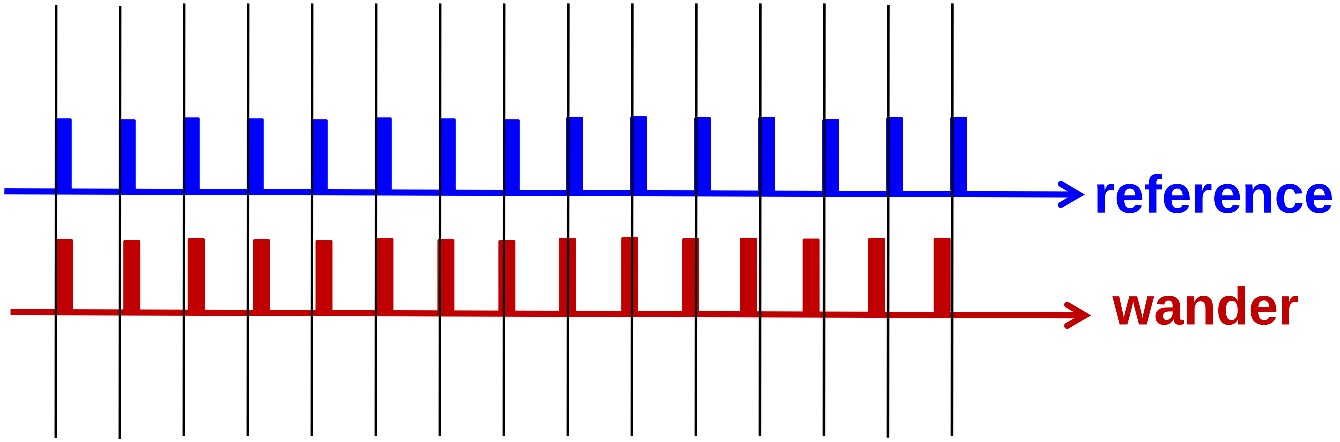


Jitter amplitude is measured in Ui_{pp} (Unit Interval peak-to-peak)

- For example – for an E1 : 1 UI_{pp} = 1/2MHz = 488 ns

Frequency distribution **wander**

Slow meandering of TIE is called wander



For TDM systems, the jitter-wander dividing line is 10 Hz

Two widely used wander measures are **MTIE** and **TDEV**

MTIE is the maximum peak-to-peak variation of TIE
in all observation intervals of duration τ during the measurement

TDEV is a measure of the expected time variation TIE
as a function of integration time, after removing FFO effects

In order to reach a required performance level
both are required to obey **masks**

The **means – frequency**

There are many ways to obtain stable and accurate frequency

The most common are :

- use of local frequency references
  - crystal oscillators
  - atomic clocks
- exploiting synchronous networks
  - TDM, SDH
  - SyncE
- exploiting wireless
  - GPS
- timing packets
  - periodic packet stream
  - time distribution protocols

Exploiting **frequency references**

Local frequency references can supply frequency for applications

All local references suffer from

- **drift** depending on environment (temperature, humidity, etc.)
- **aging**

In order of accuracy (low to high *long-term* accuracy)

- LC circuits
- piezoelectric crystal oscillators
- temperature compensated crystal oscillators (TCXO)
- crystal oscillators in temperature controlled environments (OCXO)
- cavity resonators
- Rubidium atomic clocks
- Cesium atomic clocks
- Hydrogen masers

A frequency reference which is stable/accurate enough (within 10^{-11} of UTC frequency) is called a **PRC** (Primary Reference Clock)

Exploiting TDM networks

Since highly accurate frequency references are expensive
it is usual to have only one PRC (or a small number of them)
and *distribute* its frequency to all locations where it is needed

Conventional synchronous (TDM) networks

- require accurate frequency for their own use
- automatically distribute frequency in their **physical layer**
- distribute frequency end-to-end by master-slave relationships
- maintain a hierarchy of timing strata (PRC, stratum 1, stratum 2, …)
- each stratum level has well-defined performance parameters

This frequency can be provided as a service for other needs

Since these networks are ubiquitous
frequency distribution services are often free or inexpensive

For example, if an E1 supplies data to a GSM cell-site
its physical layer frequency can be used to lock RF

A TDM alternative - **SyncE**

Asynchronous (Ethernet/MPLS/IP) networks
are rapidly replacing synchronous networks
Free frequency distribution is thus becoming much rarer

But there is a way of having your cake and eating it too

The standard Ethernet physical layer is not frequency locked
but it is easy to replace it with a synchronous one

This is the idea behind **SyncE** (Synchronous Ethernet)

SyncE does not change packet performance

- the physical symbol rate is made synchronous
- but Ethernet frames are still released asynchronously

End-to-end frequency distribution requires end-to-end support
for existing networks this may require *forklift upgrade*

Exploiting **wireless**

Another ubiquitous frequency-carrying physical layer is **wireless**

The US **GPS** satellite system

- transmits a highly accurate (*long term*) frequency reference
- covers over majority of the world's surface
- can be received as long as there is a clear view of the sky
- has built-in redundancy
- receiver price is now minimal

Similar to GPS :

- the Russian **GLONASS** system
- the Chinese **COMPASS** (Beidou) system
- the new European **Galileo** system

The **LORAN** low frequency terrestrial Maritime navigation system can be similarly used

For some applications, local accurate RF can be *appropriated*

Exploiting **packet traffic**

If there is no continuous physical layer carrying periodic events then we must distribute frequency as *information* (data)

A simple method is to send a *periodic* stream of **timing packets**

However, while these packets are sent at a rate R
that is, at times $T_n = n R$

they arrive at times $t_n = T_n + d + V_n$ where

- D = average propagation delay through the network
- V_n = **PDV** (Packet Delay Variation)

But, by proper averaging/filtering (actually control loops are needed)

$$<t_n> = T_n + d = n R + d$$

and the packet rate R has been recovered (d is unimportant for now)

This is called **ACR** (Adaptive Clock Recovery)

Note : the filtering takes (convergence) time (uncertainty theorem)

Packet frequency distribution

While ACR often works well, it has disadvantages

- changes in the average delay (rerouting) must be detected
- PDV may be easy or hard or impossible to filter out
  - easy when it is highpass noise (jitter)
  - minimum-gating eliminates PDV for small number of switches
  - hard for slow drifting changes (e.g., queuing effects)
  - impossible when the PDV looks like oscillator wander

And although ACR may work 99% of the time
there can be **no guarantees** for all network conditions
for example, extended unavailability or faults in higher layers

Another problem in practice is *frequency beating*
when unlocked timing streams traverse a single switch

Note that **PLR** (Packet Loss Ratio) is not normally a problem
ACR usually *steers* (disciplines) a low jitter local oscillator
so the problem is never jitter – it is only filtering out the wander
and Nyquist sampling wander does not require a high packet rate !

Use of **timestamps**

Instead of sending a periodic stream of timing packets
  we can send a nonperiodic stream
  but insert into the packet a **timestamp**
  identifying the precise time the packet was sent
  if this is hard to do accurately, we can use a follow-up packet

This functionality exists in all time distribution protocols

For *frequency distribution*, we only need a one way protocol
Packets sent (broadcast/multicast/unicast)

- from master fed by PRC
- to slave requiring accurate frequency

Time distribution requires a 2-way dialog (ranging)

On-path support elements

Sometimes we can improve packet-based timing distribution by using special functions distributed throughout the network

This is cheating, but some timing distribution protocols can exploit such elements if they are there

Examples :

LINK SUPPORT

1) a SyncE link
2) a POS path with frequency available to user
3) a DSL link with NTR

NETWORK ELEMENT SUPPORT

1) a network element with local frequency (e.g. atomic clock)
2) a network element with local time (e.g. GPS)
3) boundary clock
4) transparent clocks

Frequency – accuracies

What frequency performance levels can be expected in practice ?

Physical layer distribution

- GPS : PRC long term, 100s of ppb short term
- SDH : stratum levels, < 5 ppb short term
- SyncE networks behave exactly as SDH networks

Packet frequency distribution

- with on-path support - best case* : similar to SDH
- 10 noncongested switches without support : < 50 ppb
- public Internet : < 1 ppm

* problems arise when on-path support or higher layers fail

The **means –uncalibrated time**

Time distribution requires something new

To have an event occur simultaneously with a reference event we need to know how long it took for the information on the reference event to arrive

This is called **ranging**

For physical links this involves TDR (Time Domain Reflectometry)

For packet interchange ranging requires a 2-way protocol
- master – slave master controls slave
- client – server client sends request to time server when it needs

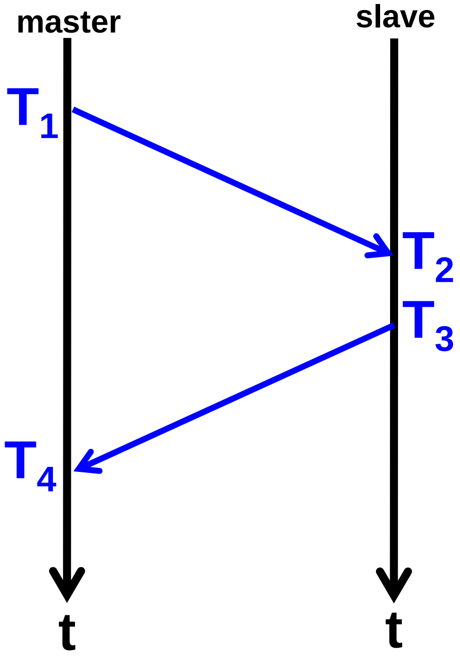
PONs (Passive Optical Networks) distribute uncalibrated time because upstream traffic is TDMA

OLT is a master to ONT slaves
- GPON locks ONT frequency and OLT compensates for time offset
- EPON locks ONT time

How is ranging performed ?

The idea behind ranging is simple (demonstrated for master-slave)

1. master sends a packet to slave at time T_1
2. packet is received by slave at time T_2
3. slave replies to master at time T_3
4. master receives reply time T_4



We can not compare T_1 and T_4 with T_2 and T_3

but

- $T_4 - T_1$ = round trip propagation time + slave processing time
- $T_3 - T_2$ = slave processing time

so

- $T = (T_4 - T_1) - (T_3 - T_2)$ = round trip propagation time

If we can assume **symmetry** ($d_{1 \rightarrow 2} = d_{3 \rightarrow 4}$)
½ T is the required one-way propagation time

If we *can not* assume symmetry
then global optimization methods are required

The frequency – time connection

Technically we do not have to distribute frequency in order to distribute (uncalibrated) time

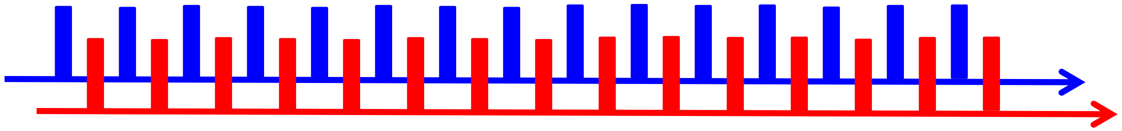
We *could* simply send frequent time updates

However, the following is more efficient :

1. slave first acquires frequency from the timing packets
2. stable and accurate 1 pps train is locally generated
3. ranging is performed
4. 1 pps train is moved to proper position

This reduces timing packet rate (and network load) but introduces a convergence time

If we have an alternate source of frequency (e.g., SyncE) then we can reduce the packet rate without convergence time



The **means –Time of Day**

Unlike frequency distribution which is for economic reasons
ToD distribution can not be avoided when ToD is needed

The simplest way (e.g., IETF TIME protocol) just sends a timestamp but this will be off by the time of flight (can be 100s of ms)

So we need to perform ranging to obtain uncalibrated time (1 pps) and in addition to send the identification of one of the pulses

There are *many* protocols that accomplish this

The two most important families are :

- IETF **NTP** – NTPv3, NTPv4, SNTP
- IEEE **PTP** – 1588-2002 (1588v1), 1588-2008 (1588v2)

NTP vs. PTP

While similar in many ways, there are also differences

- NTP is client-server – the server maintains no state on the client
  PTP is master-slave
- NTP has no defined hardware support
  PTP has (including TCs), and also allows follow-up messages
- NTP operates over general IP networks, including Internet
  PTP is optimized for well-engineered Ethernet networks,
  especially those with on-path support (TC, BC)
- NTP time is UTC in seconds + fractional seconds
  PTP time is TAI in seconds + nanoseconds
- NTP specifies specific algorithms
  (hybrid FLL/PL, selection, clustering, combining, disciplining, etc.)
  PTP is algorithm agnostic, but provides for profiling
- NTP automatically reduces rate as accuracy improves
  PTP has no such mechanism
- NTP slaves (but not SNTP) can track multiple masters
  PTP provides a BMCA (Best Master Clock Algorithm)
- NTP has numerous security extensions
  PTP has a symmetric key annex (ever implemented ?)

Time of Day – accuracies

What time accuracies can be expected in practice ?

The practical limitations are

- asymmetry (unavoidable)
- uncompensated queuing delays
- oscillator stability
- timestamping mechanisms

Typical accuracies :

- Direct connection : 10s of ns
- 1588v2 over a small network with TCs : 50 ns
- GPS : 100 ns
- prioritorized 1588v2 over loaded network with TCs : 500 ns
- 1588v2 traversing more general network : algorithm dependent
- NTPv4 over LAN : 10 ms
- NTPv4 over the public Internet : 100 ms

Who is working on timing ?

SDOs working on various aspects of timing

- IETF – NTP and TICTOC WGs
- ITU-T SG15 Q13
- IEEE 1588
- ATIS OPTXS-SYNC

Other SDOs for specific applications

- DOCSIS
- 3GPP
- IEEE 802.3AS
- IEC SERCOS
- TTP Time-Triggered Protocol