GÉANT
Networks • Services • People

perfSONAR supporting Lola measurements

Victor Olifer
WP6 T3 MEMBER

1st European perfSONAR User Workshop, 5-6th June, London

Public

www.geant.org

LoLa – LOw LAtency Audio Visual Streaming System

- Developed by Conservatorio G. Tartini, Trieste & Consortium GARR for musical performances through the network
- You have probably seen Lola live concerts at some TNC/GEANT or other events

- Now the WP6 T1 LoLa activity investigates the ways of network performance monitoring for Lola sessions – perfSONAR is a candidate

LoLa network requirements

- Throughput:
  - 200Mbps (30 fps)
  - 600 Mbps (60 fps)
  - > 1 Gbps
- Latency < 25 -35 ms
- Jitter < 3 ms
- Packets loss < 0.3 %

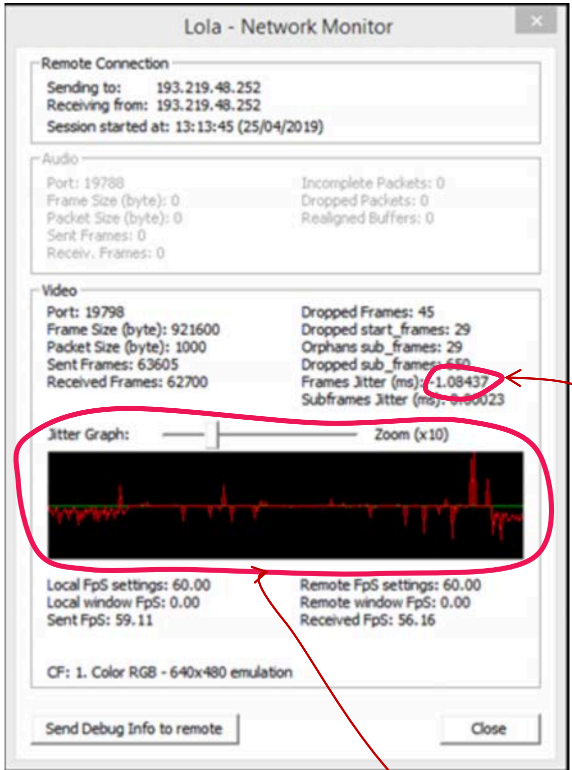
Not always the case, especially when Lola traffic runs

Delay & Jitter tolerance depends on musicians experience

Monitoring is needed before the concert goes live to check whether network paths between sites are good enough

LoLa Tester – LoLa's own performance monitoring tool

It can work on LoLa machines or on separate ones – but only Windows



1. Tester-sender generates/emulates Lola video traffic with required FPS & resolution parameters

2.Tester-receiver registers the time of every Lola frame arrival with its own clock

3. Tester-receiver calculates difference between frame arrival time and expected arrival time, calculates average difference every second and shows it on screen

No jitter data are stored on disk, screenshot is the only way to have some evidence – quite inconvenient

Tester-receiver shows jitter graph on screen but without scale

perfSONAR jitter measurements

- Measures delays, calculates two jitter statistics, stores them but … doesn't display (yet) on web page, delays are main focus

- Shows jitter and delay histogram through CLI:

  - [victor@ps-slough ~]$ pscheduler task latency --dest 193.219.48.249 --source ps-slough-1g.ja.net
  - Max Clock Error …… 0.39 ms
  - **Common Jitter Measurements:**
  - **P95 - P50 …….. 0.04 ms**
  - **P75 - P25 …….. 0.02 ms**
  - **Variance ……… 0.00 ms**
  - **Std Deviation …. 0.05 ms**

Histogram:

23.81 ms: 1 packets
23.84 ms: 1 packets
23.85 ms: 2 packets
23.87 ms: 1 packets
23.90 ms: 1 packets
23.93 ms: 1 packets
23.94 ms: 1 packets
23.95 ms: 1 packets
23.96 ms: 1 packets
23.97 ms: 1 packets
23.98 ms: 2 packets
24.00 ms: 2 packets
24.01 ms: 2 packets
24.02 ms: 13 packets
24.03 ms: 27 packets
24.04 ms: 23 packets
24.05 ms: 8 packets
24.06 ms: 5 packets
24.07 ms: 3 packets
24.08 ms: 2 packets
24.10 ms: 2 packets

One of the WP6 T1 LoLa activity objectives :

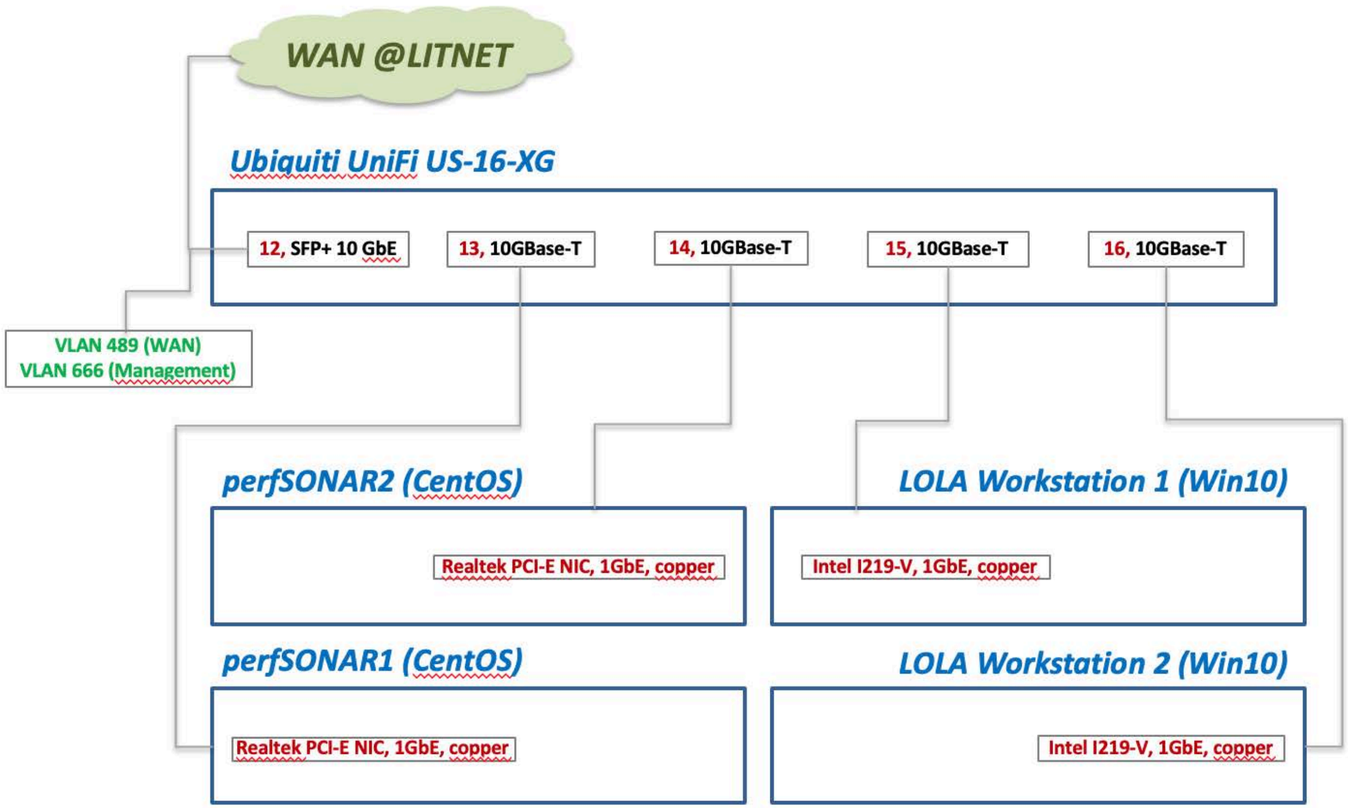
- Investigate whether perfSONAR can be used for performance monitoring of network paths for LoLa

Test plan (from Claudio Allocchio & Xavier Jeannin):

- Set up several sites with LoLa & pS machines in each site
- “Certificate” sites locally to be sure that both LoLa and pS machines are good enough – and obtain some experience with jitter measurements with LoLa tester & Ps - this is the current stage of the project
- Carry out wide area site to site tests measuring jitter by both LoLa Tester & pS using different video settings. pS should measure traffic during Lola sessions, site-to-site tests start next week
- Compare results and make conclusions

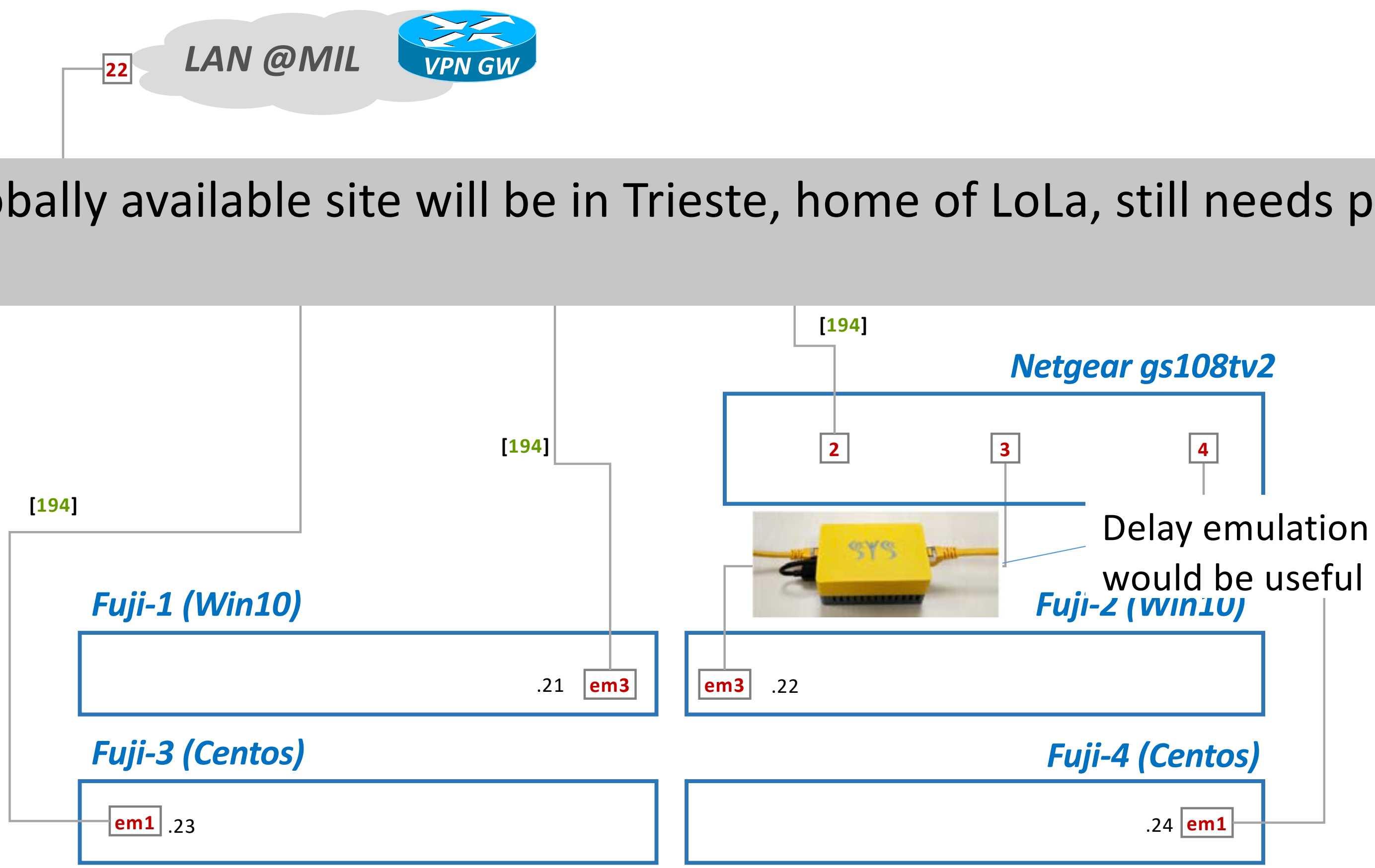
Vilnius site

(under Vytenis Gadliauskas supervision, ready for site to site tests)



Milan site

(under Fabio Farina supervision, inside of VPN and for local use only)



The second globally available site will be in Trieste, home of LoLa, still needs pS server

Do pS and LoLa Tester measure the same? A bit (5 slides) of theory

Jitter could mean different things:

From RFC 3393 IP Packet Delay Variation Metric:

"Jitter" commonly has two meanings:

- The first meaning is the **variation of a signal with respect to some clock signal**
  –> close to how Lola calculates jitter, clock signal is expected interval

- The second meaning has to do with the **variation of a metric** (e.g., delay) with respect **to some reference metric (e.g., average delay** or minimum delay). This meaning is frequently used by computer scientists and frequently (but not always) refers to variation in delay -> **IPDV** (Inter Packet Delay Variation)

–> close to how OWAMP/TWAMP calculates jitter

NTP jitter: Standard Delay Deviation

```
*ntp4.ja.net .DCFa. 1 u 51 1024  377 2.880 -0.121  0.343
```

Are these ways comparable ???

Closer look at RFC 3393 definition of ipdv as a random variable

RFC 3393: The **IPDV** is the difference between the one-way-delay of the selected packets.

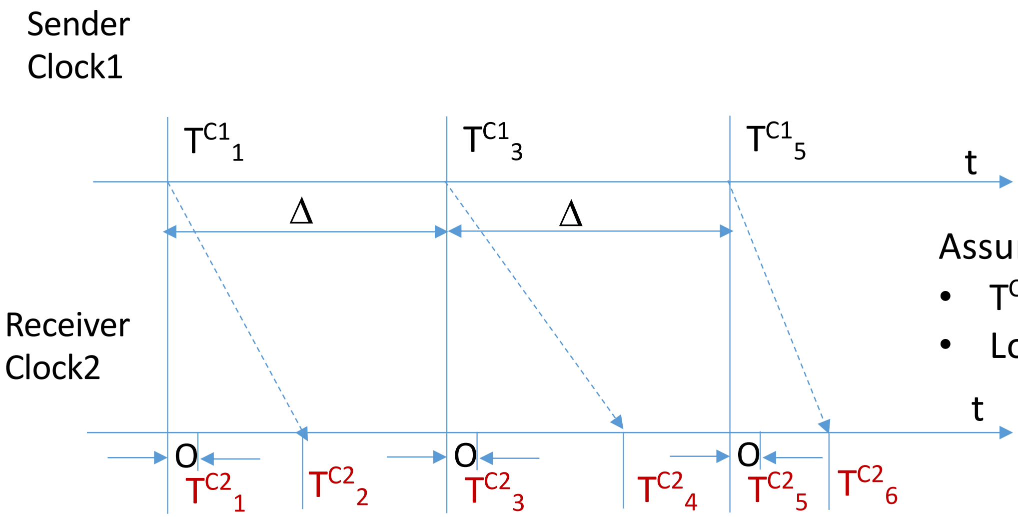
```
       I1  P(i)      P(j)                P(k)                    I2

MP1  |--------------------------------------------------------------|
         |\         |\                   |\
         | \        | \                  | \
         |  \       |  \                 |  \
         |   \      |   \                |   \
         |dTi \     |dTj \               |dTk \
         |<--->v    |<--->v              |<--->v

MP2  |--------------------------------------------------------------|

      I1          P(i)      P(j)                P(k)             I2

                   Figure 1: Illustration of the definition
```

Then ddT = dTk - dTi as defined above. -> jitter

RFC 3393 suggests but not dictates statistics for IPDV, for example:

- 50% percentile of ipdv
- Absolute values of IPDV: j_new = 15/16* j_old + 1/16*j_new

LoLa approach is very close to RFC 3393



Assumptions:

- $T^{C1} - T^{C2}$ = O(ffset) = const during measur. interval 1 sec
- Lola Tester sends packets strictly evenly, Δ = const

RFC 3393:

$$ipdv1 = \underbrace{(T^{C2}_4 - T^{C1}_3)}_{d2} - \underbrace{(T^{C2}_2 - T^{C1}_1)}_{d1} = [T^{C2}_4 - (T^{C2}_1 - O + \Delta)] - [T^{C2}_2 - (T^{C2}_1 - O)] = T^{C2}_4 - T^{C2}_1 + O - \Delta - T^{C2}_2 + T^{C2}_1 - O = \boxed{T^{C2}_4 - T^{C2}_2 - \Delta}$$

same value!!!

LoLa:

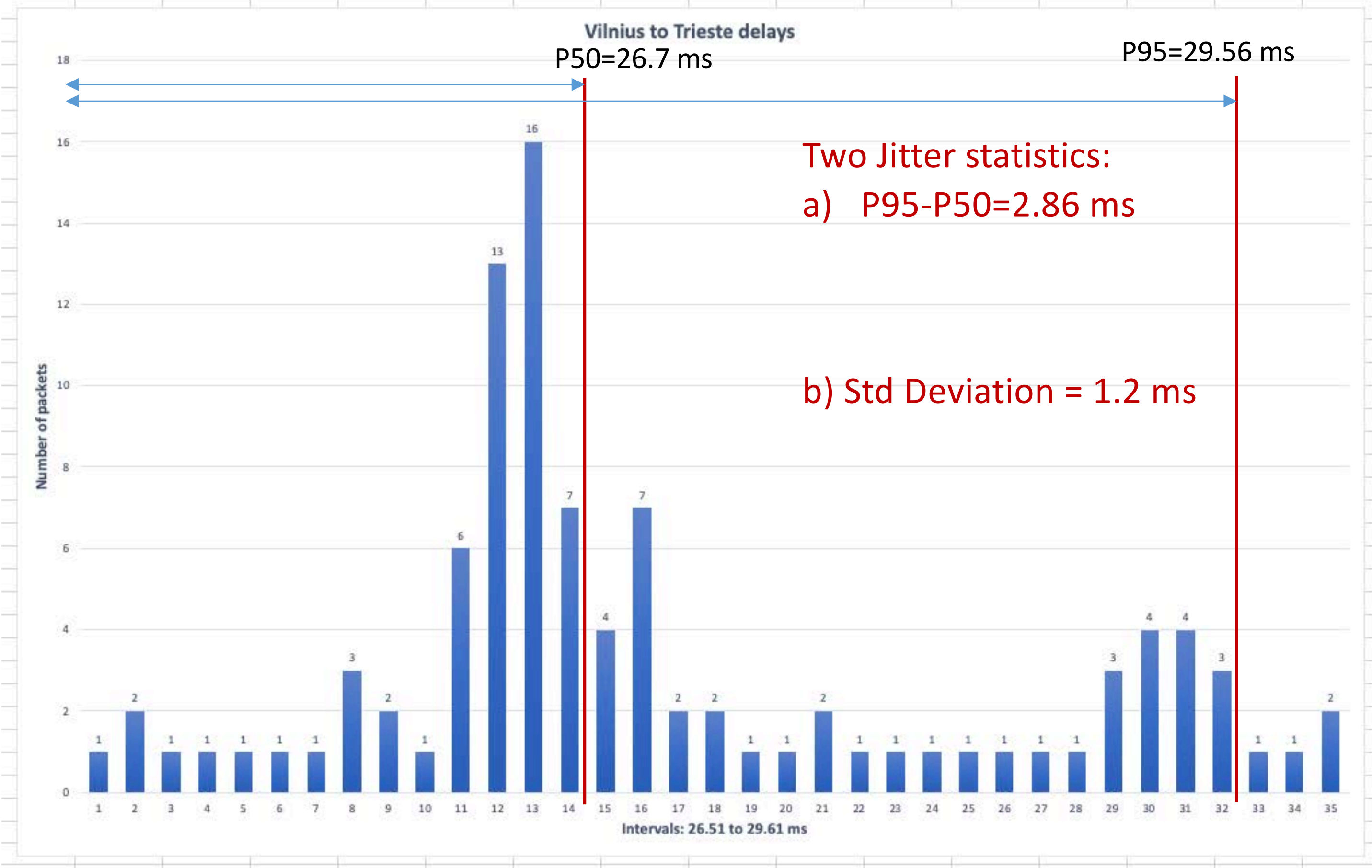
$$jitter1 \boxed{= (T^{C2}_4 - T^{C2}_2) - \Delta}$$

Conclusions:

- Lola Tester uses preliminary knowledge (Δ) and it simplifies measurements
- Lola Tester uses the same metrics as defined in RFC3393 for ipdv (when ass. true)
- Lola Tester statistics (average) is not robust, positive & negative delays can compensate each other, standard deviation would be better

What about pS measurements?

- Uses One Way Delay metric, different from RFC 3393 and LoLa Tester that measure difference in delay pairs
- But gives us Delay Histogram and two jitter statistics calculated from that Histogram and they reflect delay difference:



Let’s use the previous histogram data and calculate Lola ‘would be’ results:

1. LoLa Tester would give:

   jitter =1/100 * Σ (di – D) = 0.0421 ms, where D is average delay 27.3 ms

   -> quite far from OWAMP statistics (2.86 & 1.2 ms)

2. If Lola Tester calculated average of absolute delay variations:

   jitter would be = 0.98 ms –> closer to OWAMP statistics

3. Standard Deviation calculated from histogram as $J = \sqrt{\frac{\sum (d_i - D)^2}{N-1}}$. = 1.2 ms

-> coincides with OWAMP Std Deviation value, which shows that OWAMP uses this formula

-> Lola Tester can calculate Std Deviation from its measurements!

but is it possible to change??? Probably not

What does this theory mean for future tests?

- Now we know that LoLa Tester and pS OWAMP can produce different numbers from the same delay samples in future tests – no surprises

- Limits of OWAMP statistics & LoLa Tester can be different for the same ”bad” feeling of musicians - and we should find them while testing

Other suggestions?

Thank you

Any questions?

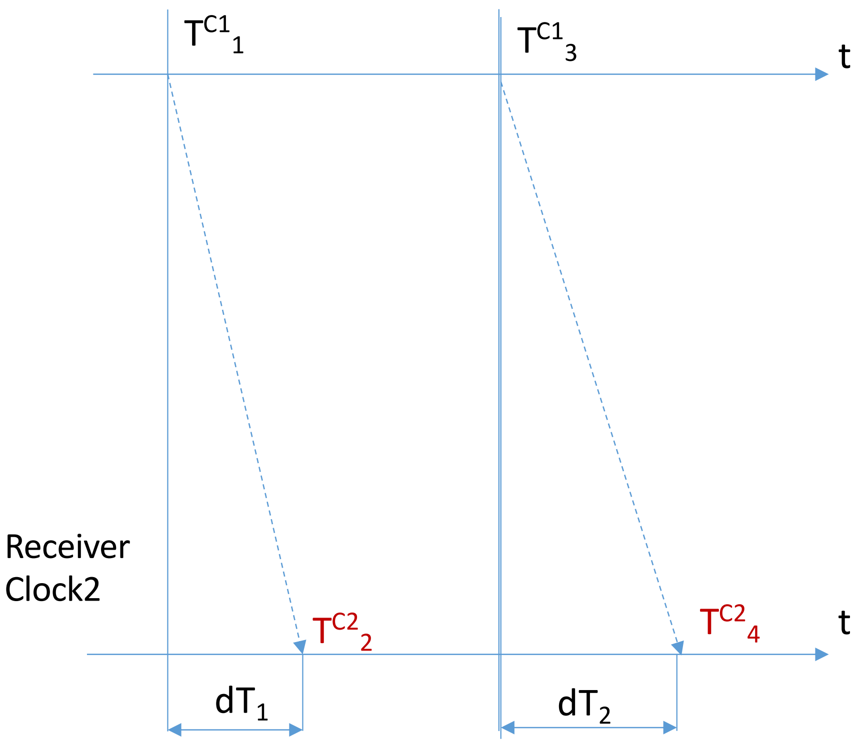
www.geant.org

© GÉANT Association on behalf of the GN4 Phase 3 project (GN4-3).
The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 856726 (GN4-3).

Extra slide - IPDV is less sensitive to clock synchronization than OWD

Sender Clock1

Case 1: $T^{C1} - T^{C2}$ = Offset = const (no skew, no drift)



1) $dT_1 = T^{C2}{}_2 - T^{C1}{}_1 = T^{C2}{}_2 - (T^{C2}{}_1 + \text{Offset})$

2) $dT_2 = T^{C2}{}_4 - T^{C1}{}_3 = T^{C2}{}_4 - (T^{C2}{}_3 + \text{Offset})$

3) $\text{IPDV} = dT_2 - dT_1 = T^{C2}{}_4 - (T^{C2}{}_3 + \text{Offset}) - (T^{C2}{}_2 - (T^{C2}{}_1 + \text{Offset})) =$

$(T^{C2}{}_4 - T^{C2}{}_3) - (T^{C2}{}_2 - T_1{}^{C2})$ -> no offset in the result!

Skew = $(f^{C1} - f^{C2})/\Delta t$ – speed of clock desynchronization – could depend on time

Drift = (Skew(t1) – Skew(t2))/ Δt – acceleration of clock desynchronization