Développement d'un Event-Builder à 10Gbps pour une caméra de télescope Cherenkov

Julien HOULES Dirk HOFFMANN

Centre de Physique des Particules de Marseille
IN2P3/CNRS

Context

Le Cherenkov Telescope Array

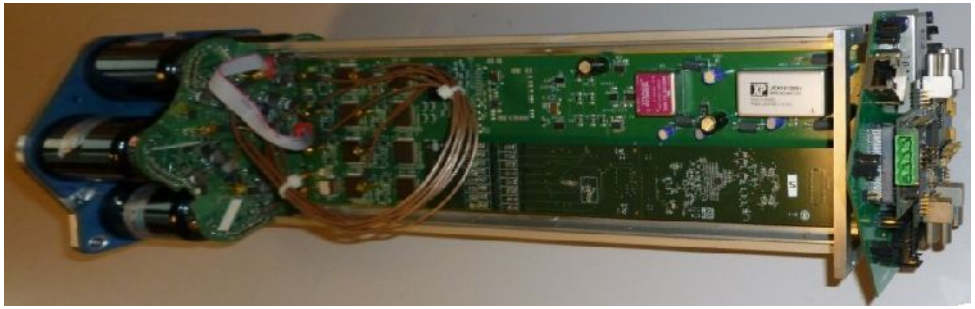
2 réseaux différents envisagés : nord et sud

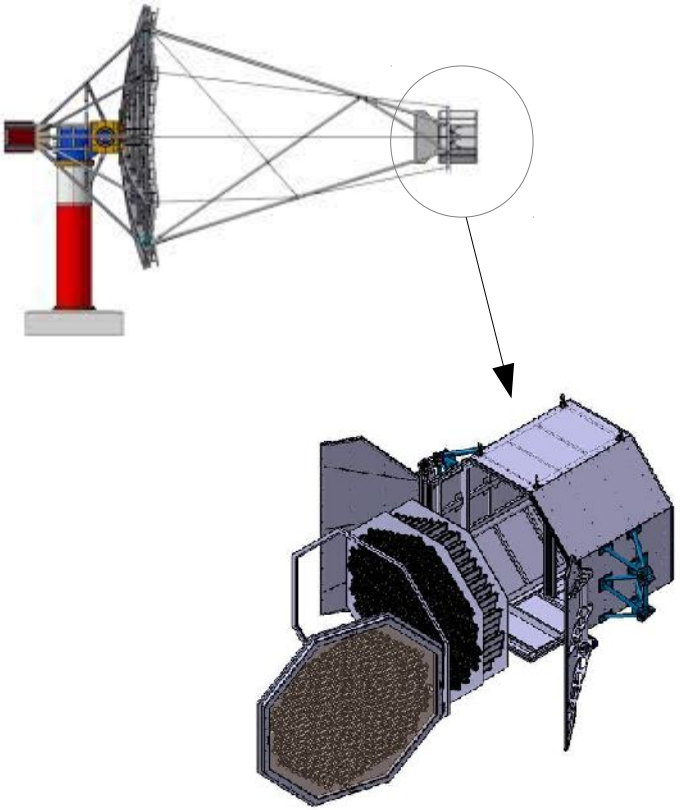
~ 100 Télescopes gamma

3 tailles différentes : LST (10 GeV-100 GeV) 24 mètres,
MST (100 GeV-1 TeV) 12 mètres,
SST (> qques TeV) 6 mètres.

Sur le MST, 2 caméras FlashCam et NectarCam

NectarCam



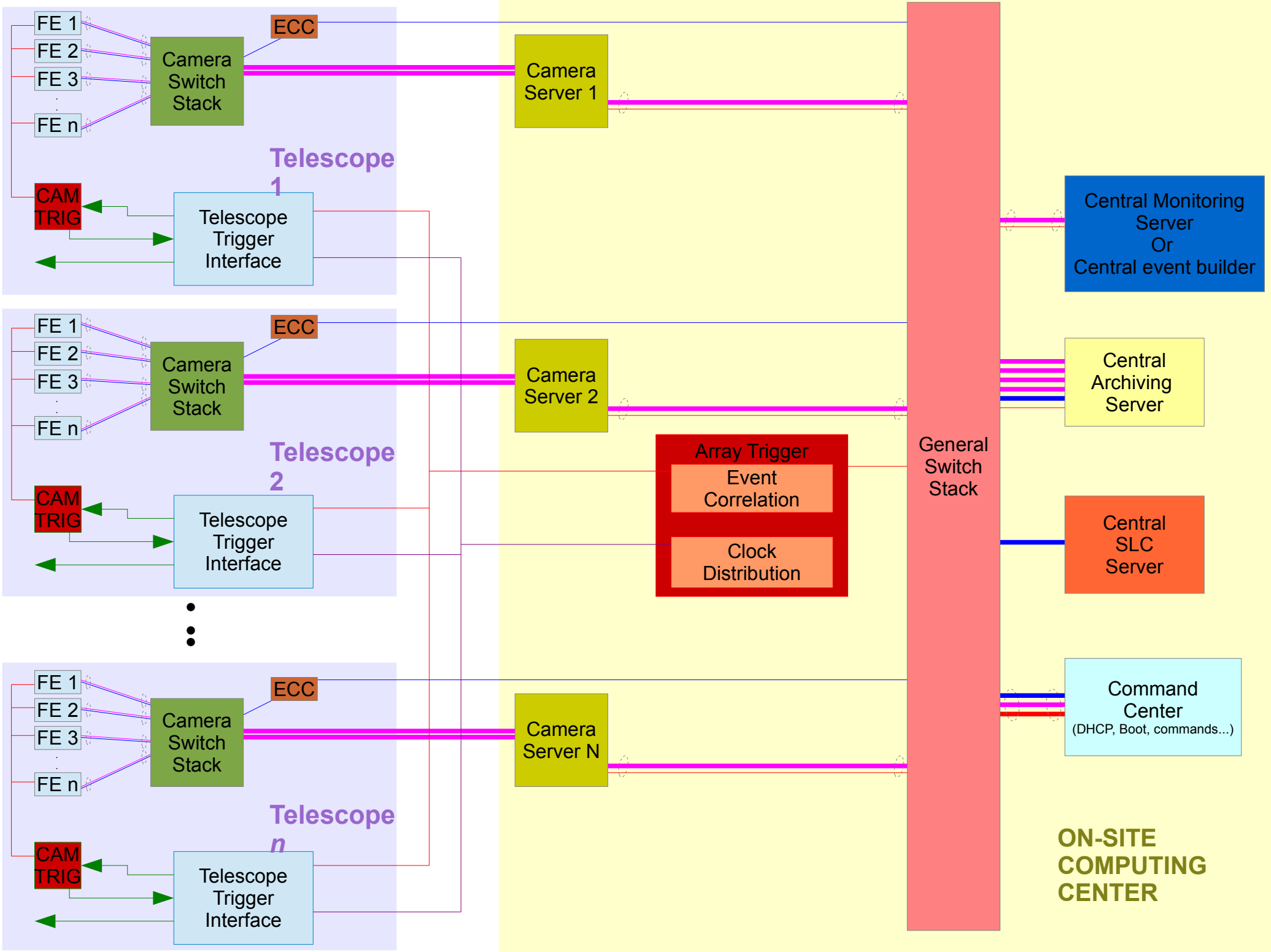


~ 1800 photomultiplicateurs (PM)

7 PM par module

→ ~250 modules

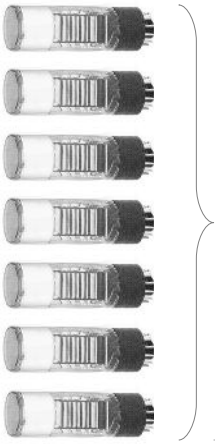
Un lien Gigabit Ethernet par module


FE 1
FE 2
FE 3
FE n
Camera Switch Stack
ECC
Camera Server 1
Telescope 1
CAM TRIG
Telescope Trigger Interface
FE 1
FE 2
FE 3
FE n
Camera Switch Stack
ECC
Camera Server 2
Telescope 2
CAM TRIG
Telescope Trigger Interface
Array Trigger
Event Correlation
Clock Distribution
FE 1
FE 2
FE 3
FE n
Camera Switch Stack
ECC
Camera Server N
Telescope n
CAM TRIG
Telescope Trigger Interface
General Switch Stack
Central Monitoring Server Or Central event builder
Central Archiving Server
Central SLC Server
Command Center
(DHCP, Boot, commands...)
ON-SITE COMPUTING CENTER

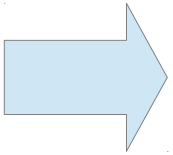
Front end data flow

Hypotheses

PM



Front end board



event 1	event 2	...	event N
Header	Header		Header
PM 1 HG/LG	PM 1 HG/LG		PM 1 HG/LG
PM 2 HG/LG	PM 2 HG/LG		PM 2 HG/LG
PM 3 HG/LG	PM 3 HG/LG		PM 3 HG/LG
PM 4 HG/LG	PM 4 HG/LG		PM 4 HG/LG
PM 5 HG/LG	PM 5 HG/LG		PM 5 HG/LG
PM 6 HG/LG	PM 6 HG/LG		PM 6 HG/LG
PM 7 HG/LG	PM 7 HG/LG		PM 7 HG/LG

Whole Camera ~ 2016 PM -> 288 front end boards

Camera DAQ Hypotheses

Hypotheses

- 2016 pixels camera

- Average camera trigger rate : 8 kHz
- Maximum size of waveforms for 1 PM :
  144 bytes (16 bit * 2 gains * 36 samples)

→ Max theorical bandwidth = 8000 * 2016 * 144 = 2.32 GB/s
= 18.6 Gb/s

- 7 detectors for each front end board : 288 boards/camera

→ Each board generates a flow of 2320/288 = 8 MB/s
= 64 Mb/s

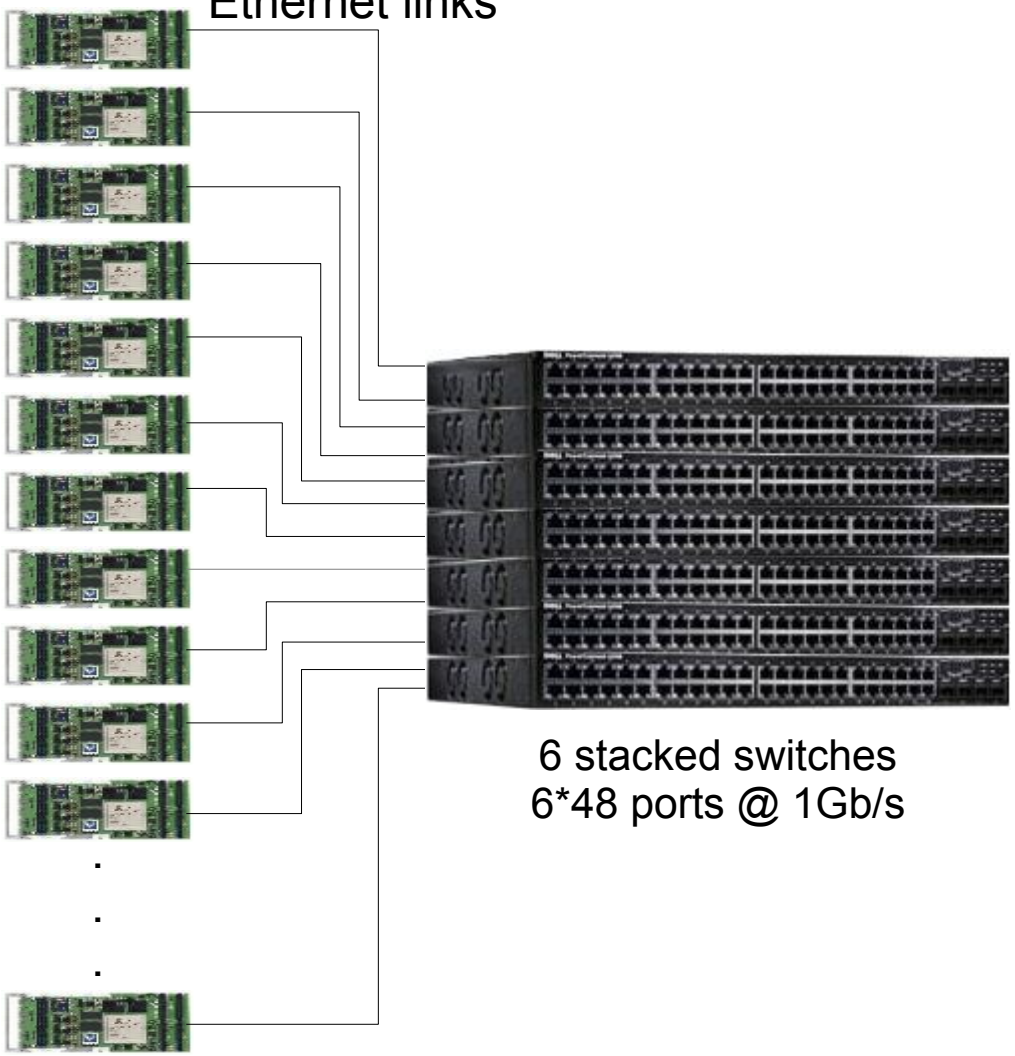
With 1024 bytes packets (network headers neglegted)

Camera DAQ Architecture

Hypotheses

288 * 1Gb/s
Ethernet links

Front end boards



6 stacked switches
6*48 ports @ 1Gb/s

2 * 10Gb/s
Ethernet links
SFP+

Camera server

Critical points

Critical points

Poissonian distribution of the triggers :
The average data rate for a trigger rate of 8 kHz can be forwarded on the two uplinks but when instantaneous rate becomes much higher, packets must be stored.

Critical points

Poissonian distribution of the triggers :
The average data rate for a trigger rate of 8 kHz can be forwarded on the two uplinks but when instantaneous rate becomes much higher, packets must be stored.

Synchronicity of the packets sending :
Will the switches be able to process packets arriving exactly at the same time on all the ports (1 ns) ?

Critical points

Poissonian distribution of the triggers :
The average data rate for a trigger rate of 8 kHz can be forwarded on the two uplinks but when instantaneous rate becomes much higher, packets must be stored.

Synchronicity of the packets sending :
Will the switches be able to process packets arriving exactly at the same time on all the ports (1 ns) ?

Packets reception and processing at link speed :
Medium-sized packets reception with regular Linux sockets at link speed is not possible

Critical points

Poissonian distribution of the triggers :
The average data rate for a trigger rate of 8 kHz can be forwarded on the two uplinks but when instantaneous rate becomes much higher, packets must be stored.

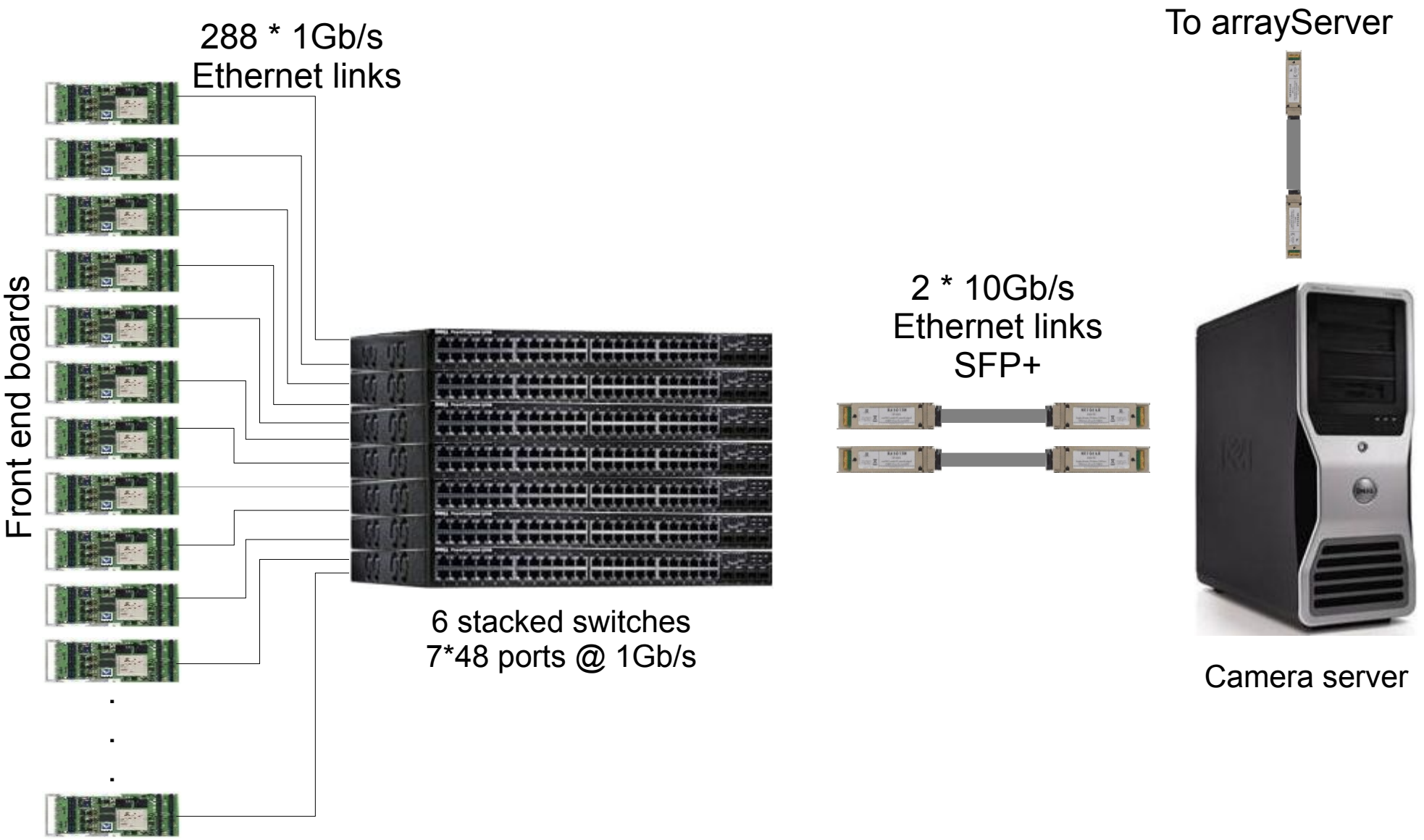
Synchronicity of the packets sending :
Will the switches be able to process packets arriving exactly at the same time on all the ports (1 ns) ?

Packets reception and processing at link speed :
Medium-sized packets reception with regular Linux sockets at link speed is not possible

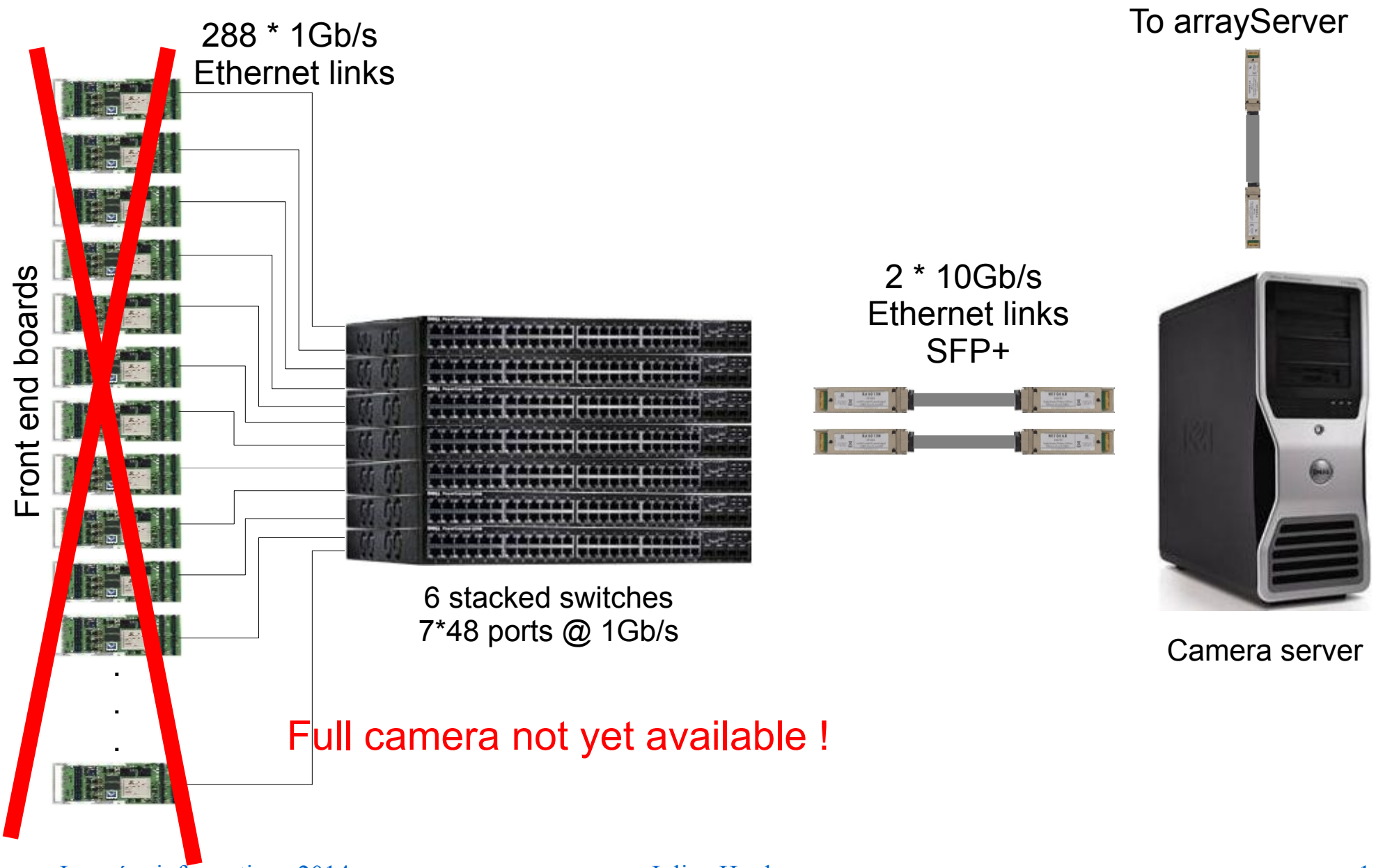
→ Need to build a stimulator able to validate the acquisition chain

Stimulator

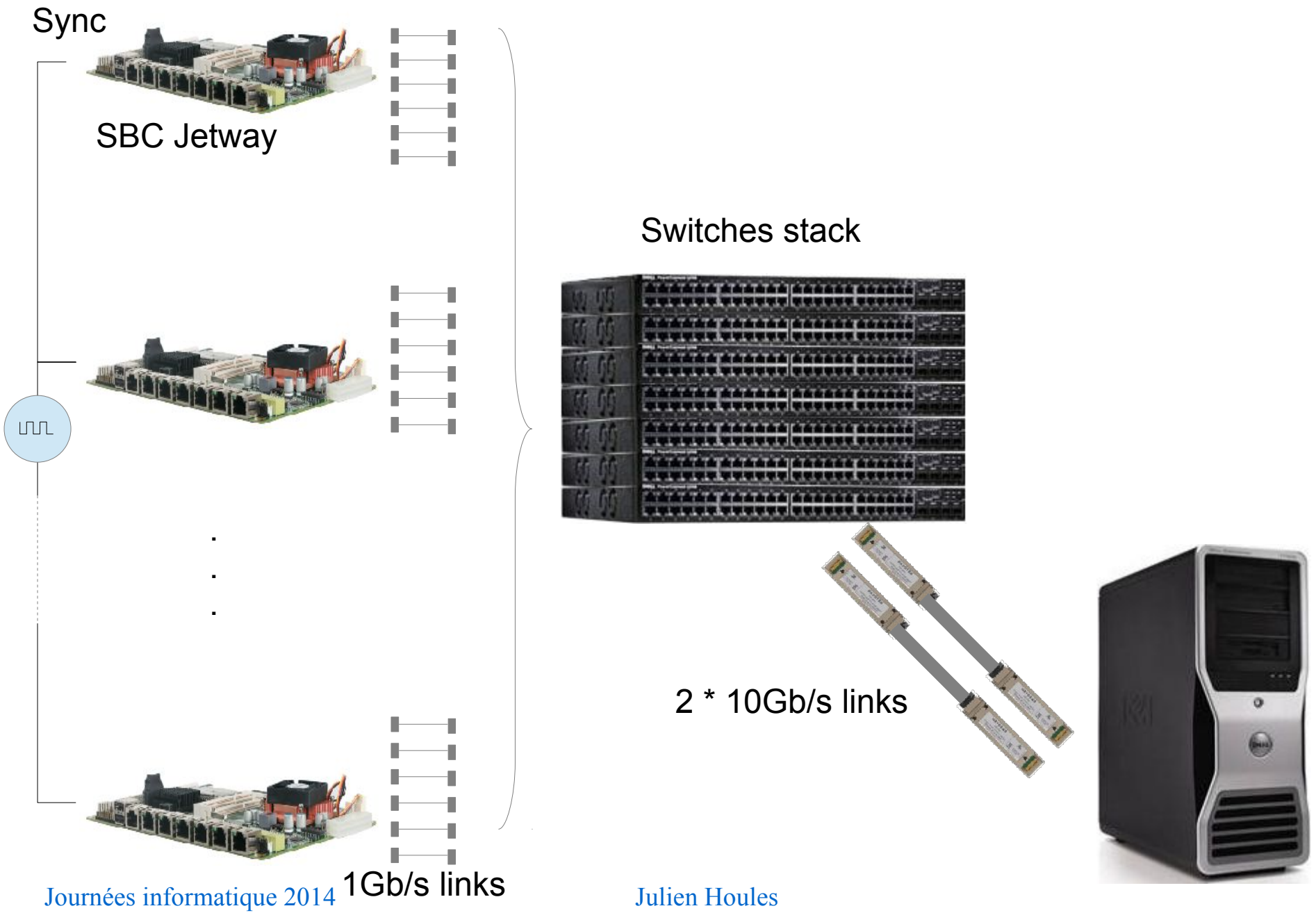
Camera DAQ Architecture



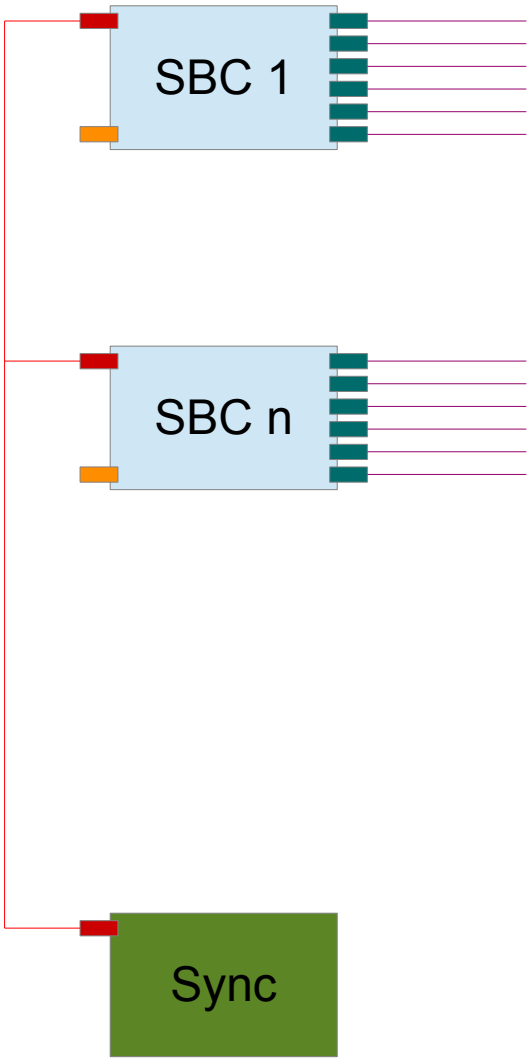
Camera DAQ Architecture

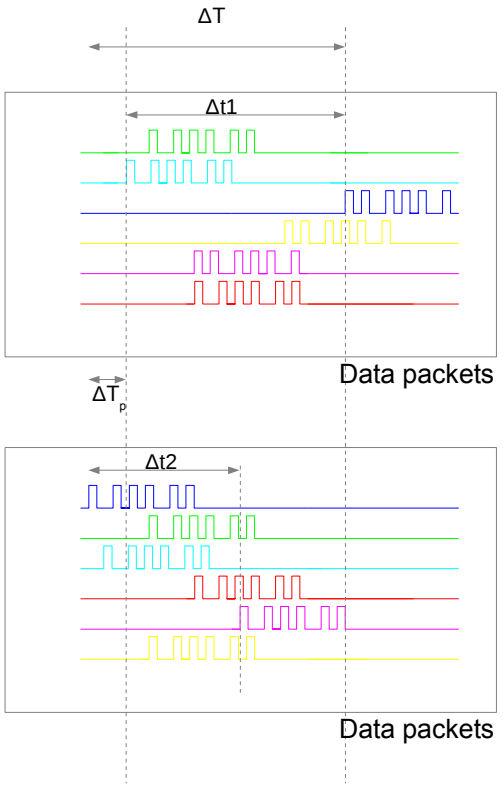


Stimulator : Architecture



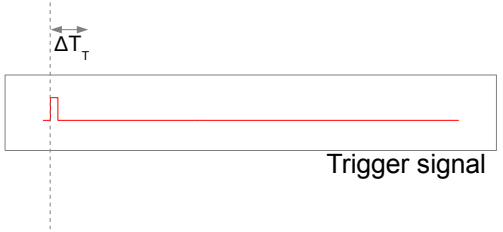
Main obstacles





Δt1, Δt2 : time difference between the first packet sent and the last one on one SBC

ΔT_p : maximum time difference between the first packets sent by each SBC



ΔT_T : time difference between the « send packet signal » and the first packet sent among all the SBCs

Main obstacles

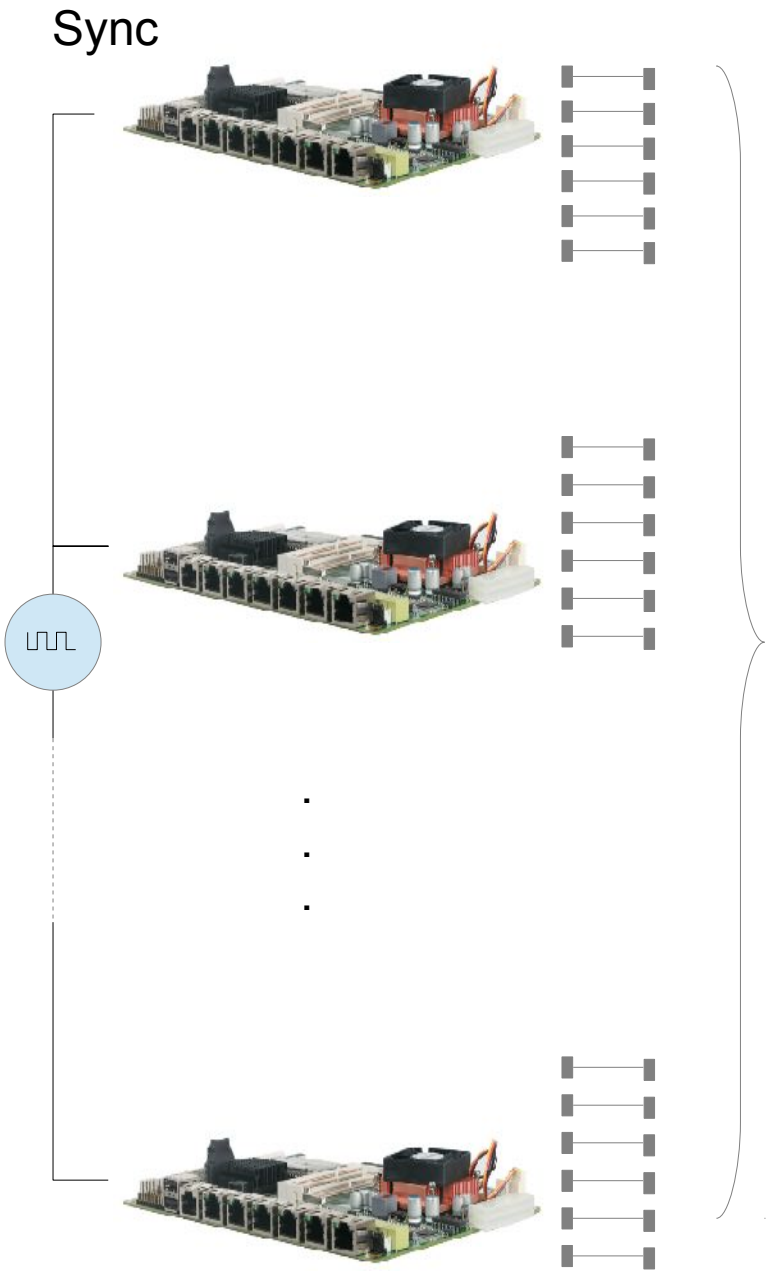
First tests with packets sending on an interrupt were very disappointing :

About 30 µs with a great jitter between the first and last paquet sent by one board and even more with two boards

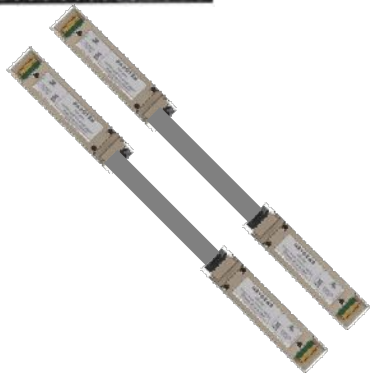
Results slightly improved with Xenomai (real time patch for Linux)

Need to avoid interrupt latency, serialize packets sending and bypass Linux communication layers to talk directly to hardware.

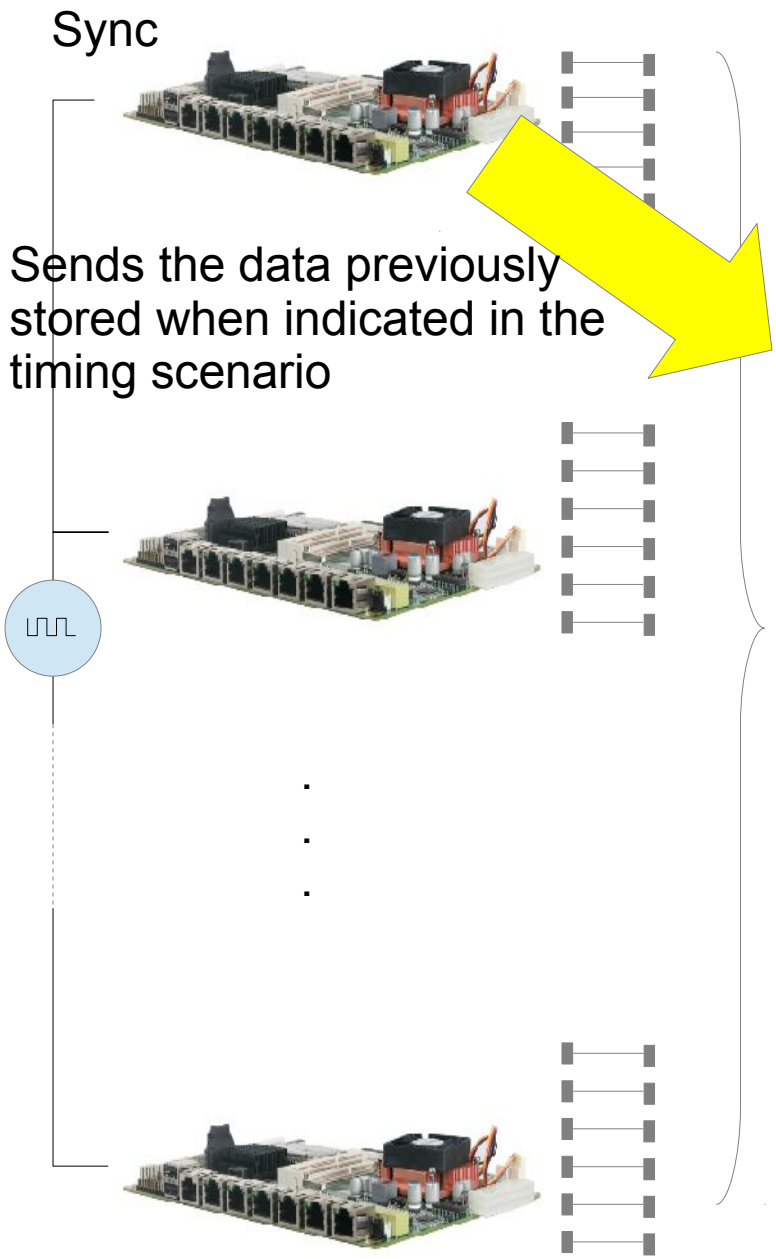
Stimulator : Architecture



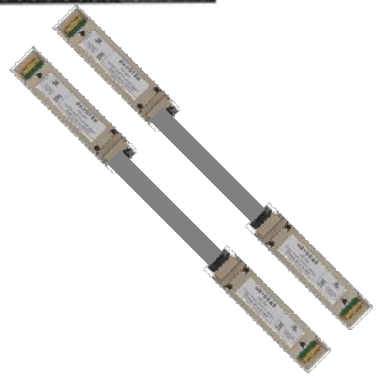
Upload :
- the scientific data simulating the FE boards replaced by the SBC board
- a timing scenario for the simulated events



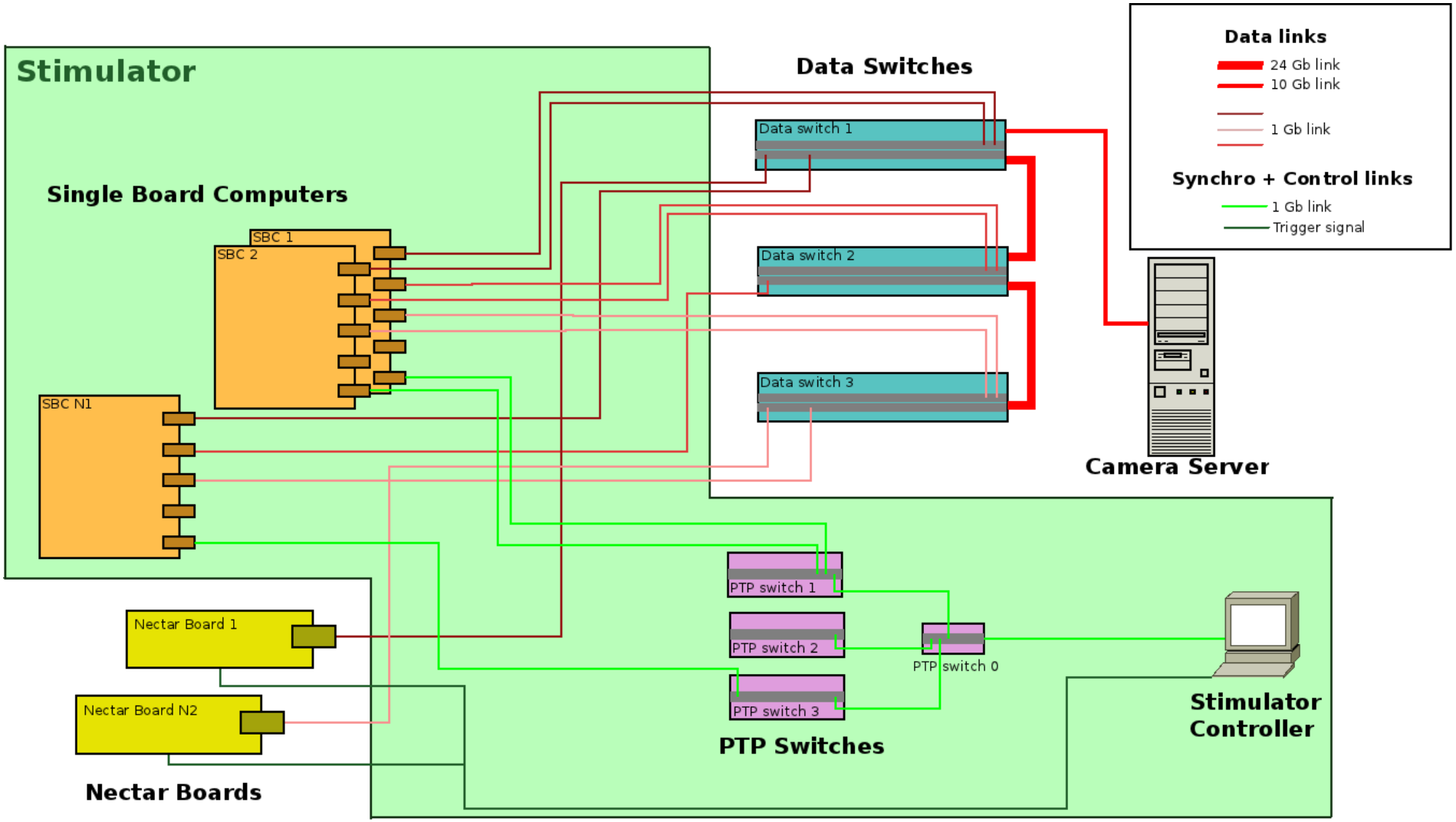
Stimulator : Architecture



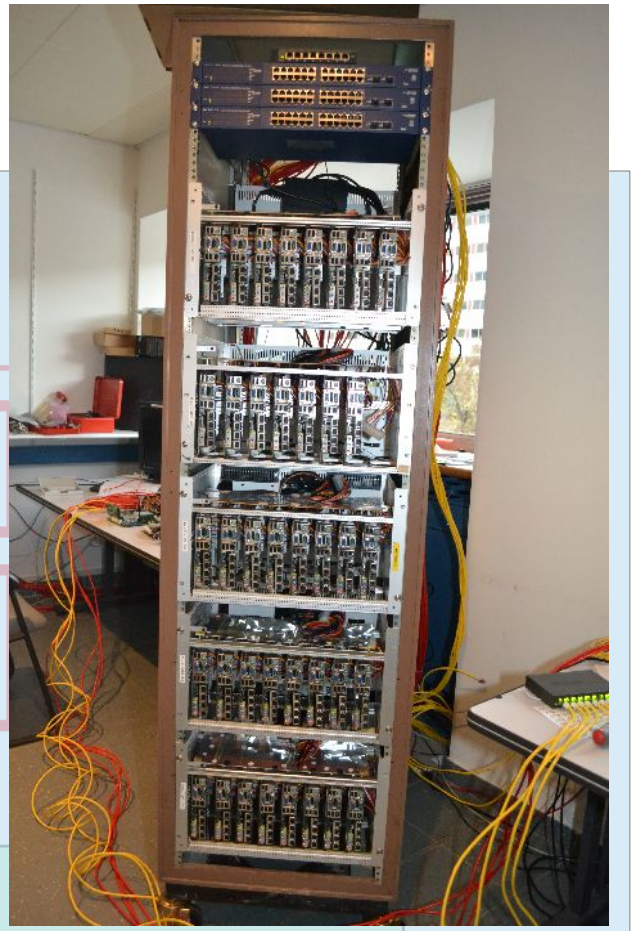
Upload :
- the scientific data simulating the FE boards replaced by the SBC board
- a timing scenario for the simulated events



Lab setup


Stimulator
Data Switches
Data links
24 Gb link
10 Gb link
1 Gb link
Synchro + Control links
1 Gb link
Trigger signal
Single Board Computers
SBC 1
SBC 2
SBC N1
Data switch 1
Data switch 2
Data switch 3
Camera Server
PTP switch 1
PTP switch 2
PTP switch 3
PTP switch 0
PTP Switches
Nectar Board 1
Nectar Board N2
Nectar Boards
Stimulator Controller

Solutions applied



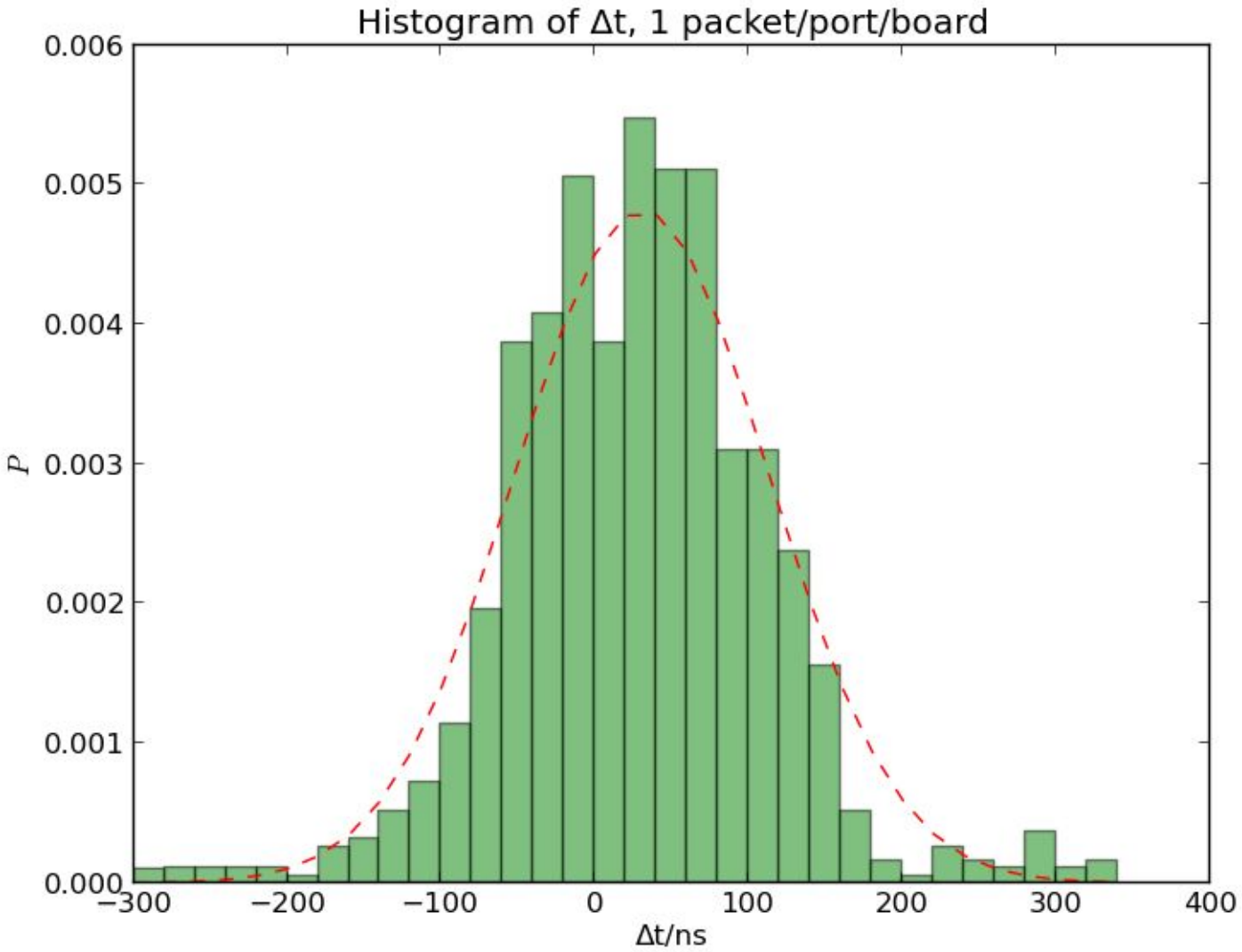
Modified PTP protocol :

Delay requests inhibited (fixed on slave side)

By-pass Linux communication layer :

Patch e1000e driver and create a new module which
- allocates custom DMA able memory buffers
- separates buffers preparation and sending triggering
- inhibits Tx interrupts
- reduces serialization to sending triggering
- sends one or more packets at a precise time : polling on real time clock

Results



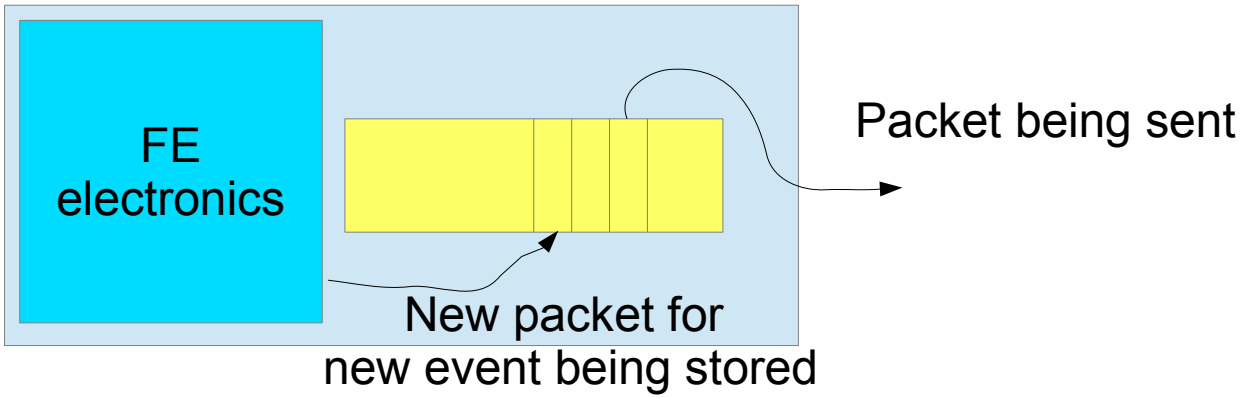
Mean = 30 ns

Sigma = 83 ns

Data bursts management

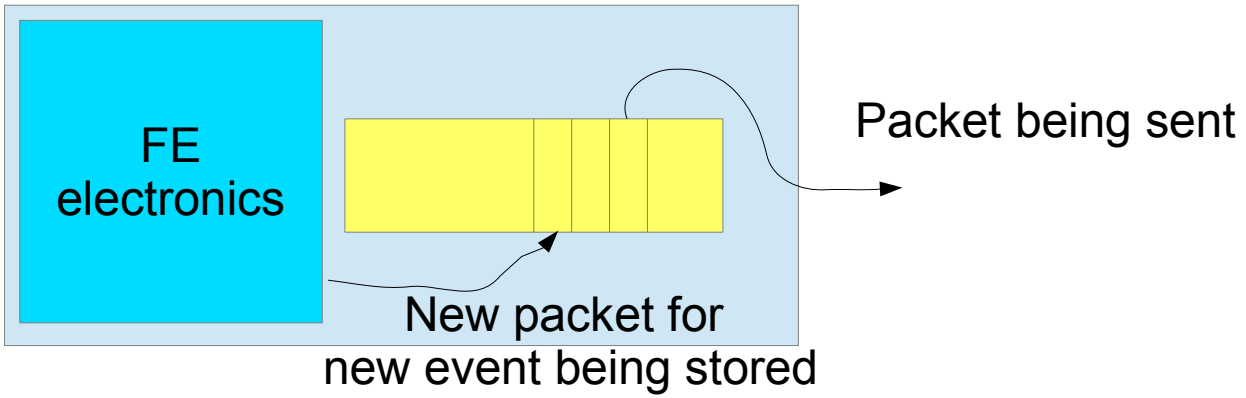
Instantaneous flow : critical point

- Average camera trigger rate : 8 kHz
- Time to send a 1024 bytes packet on a 1 gigabit link : 8 µs
- If *FE dead time* < *Δt between triggers* < *8 µs*, the packet is stored in the FE packets buffer



Instantaneous flow : critical point

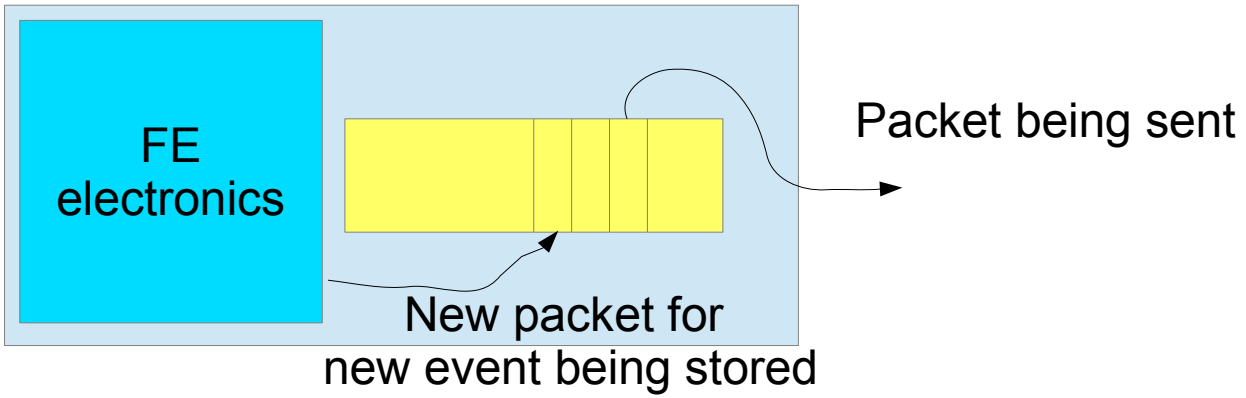
- Average camera trigger rate : 8 kHz
- Time to send a 1024 bytes packet on a 1 gigabit link : 8 µs
- If *FE dead time* < *Δt between triggers* < *8 µs*, the packet is stored in the FE packets buffer



In this case, the front end board generates a **dataflow at link speed (1Gb/s)** until its packet buffer is empty.

Instantaneous flow : critical point

- Average camera trigger rate : 8 kHz
- Time to send a 1024 bytes packet on a 1 gigabit link : 8 µs
- If *FE dead time* < *Δt between triggers* < *8 µs*, the packet is stored in the FE packets buffer



In this case, the front end board generates a **dataflow at link speed (1Gb/s)** until its packet buffer is empty.

Will the switches be able to manage such an instantaneous flow **without losing packets** ?

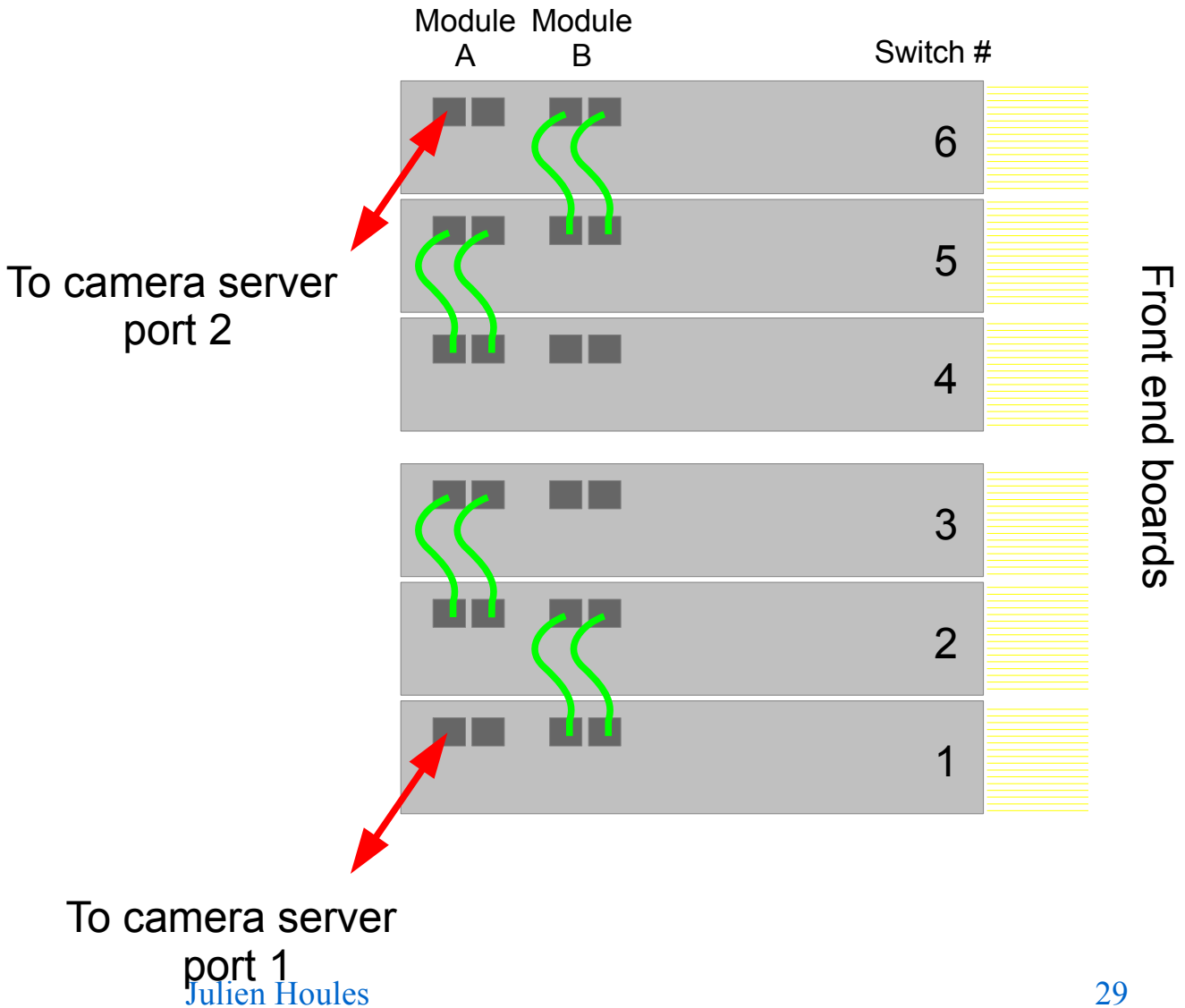
Switches stack links

Switches stack rear panel :

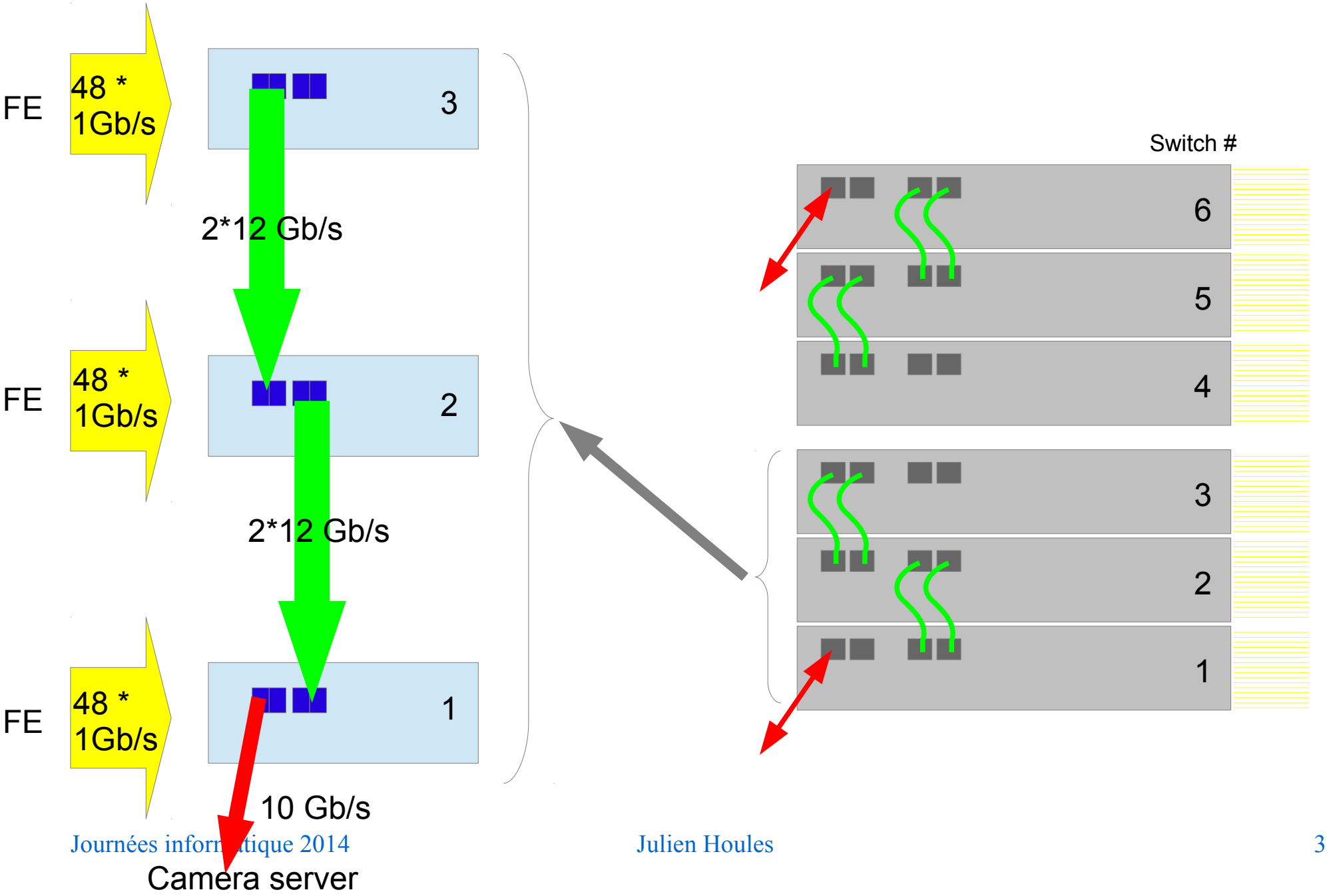
12 Gb stacking links

10 Gb uplinks

1 Gb FE links

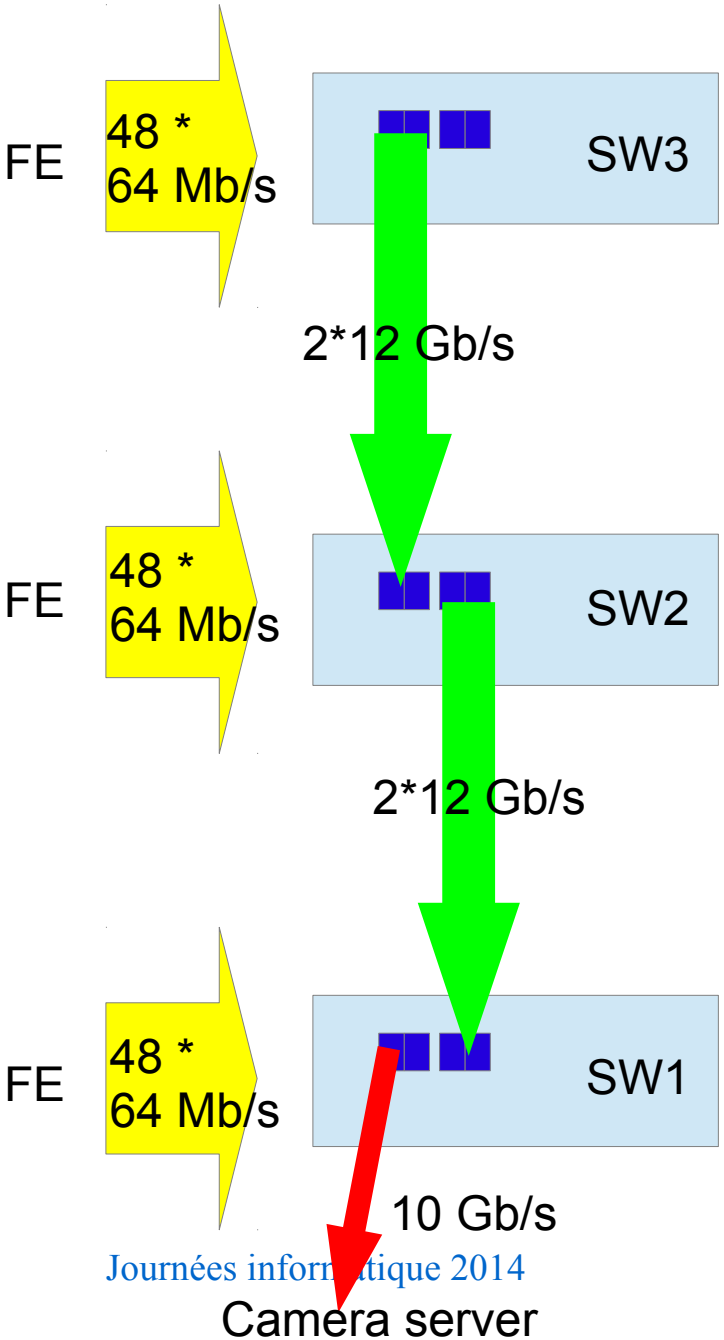


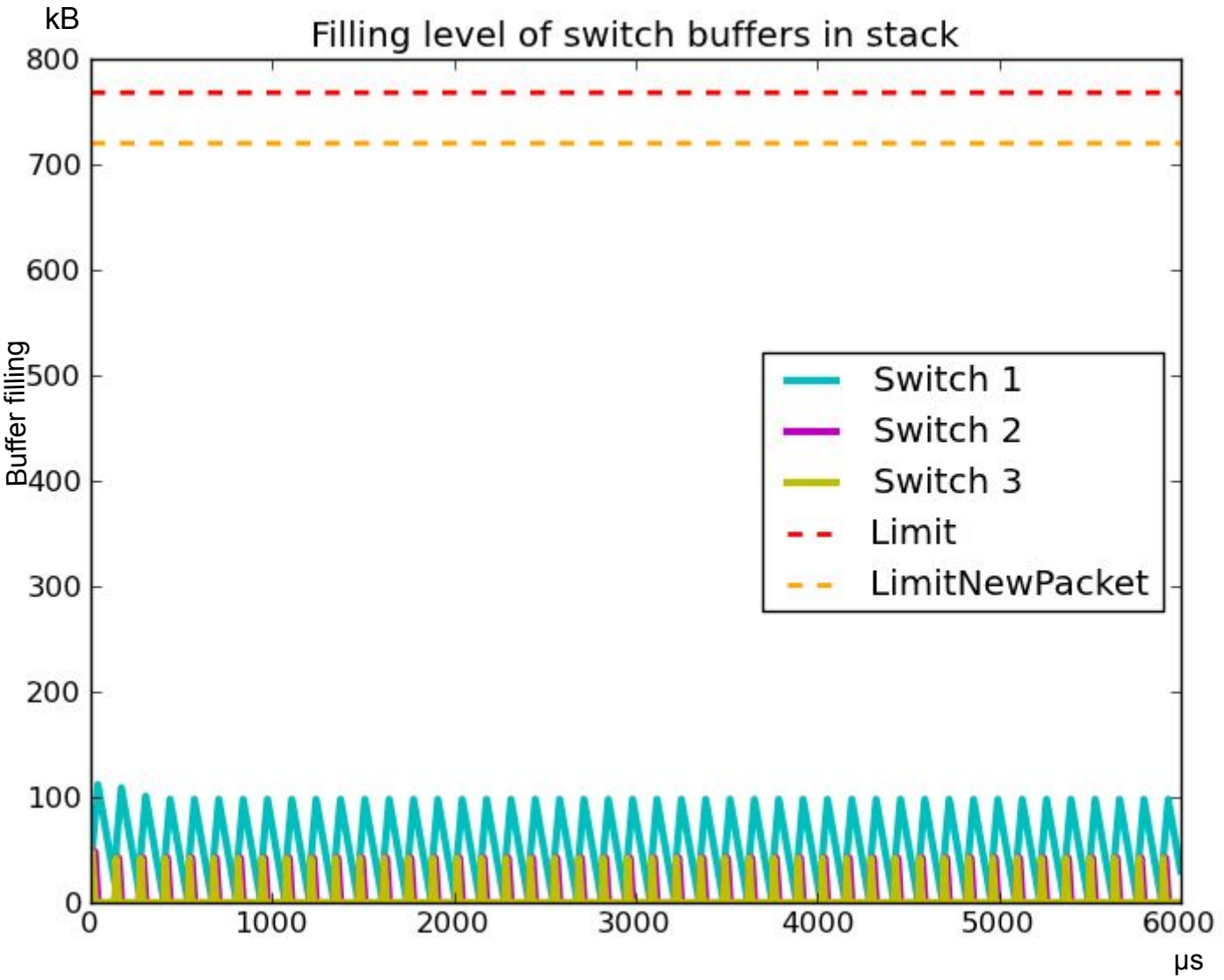
Switches stack model



Switches stack model

Each switch has a capacity of **768 kB**

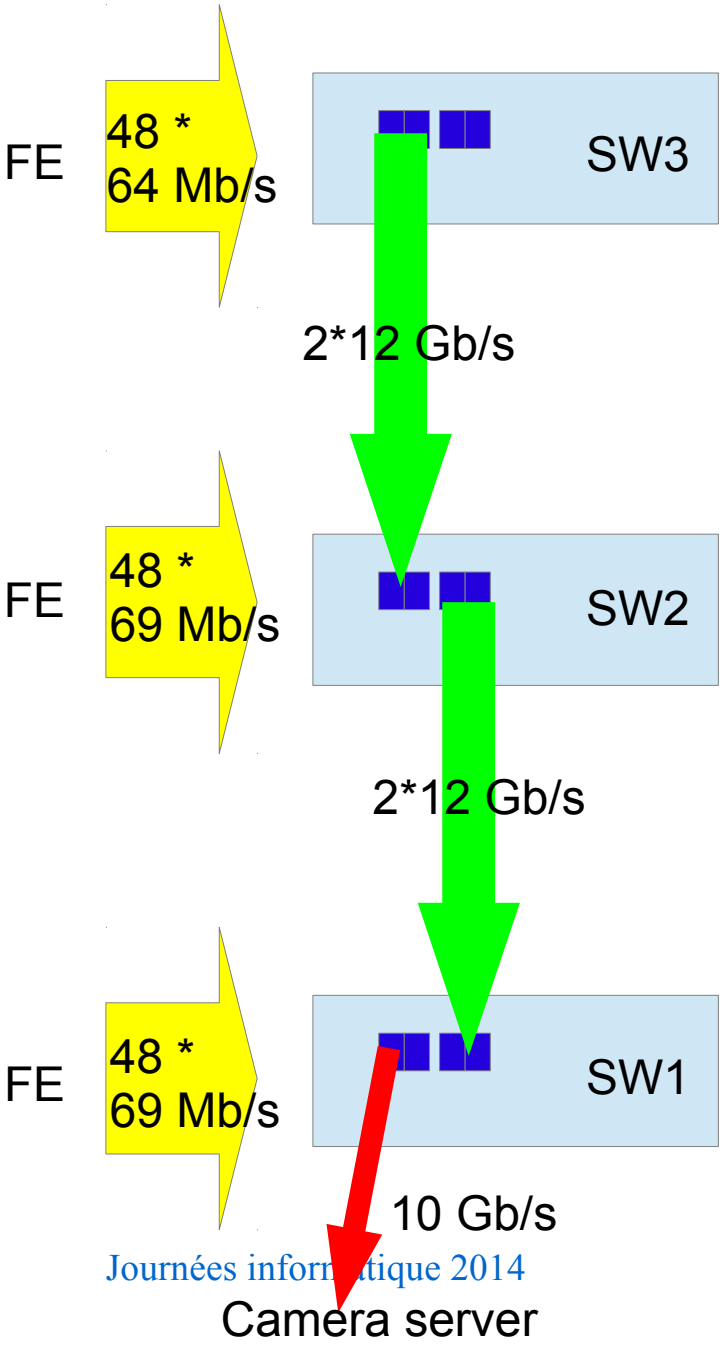


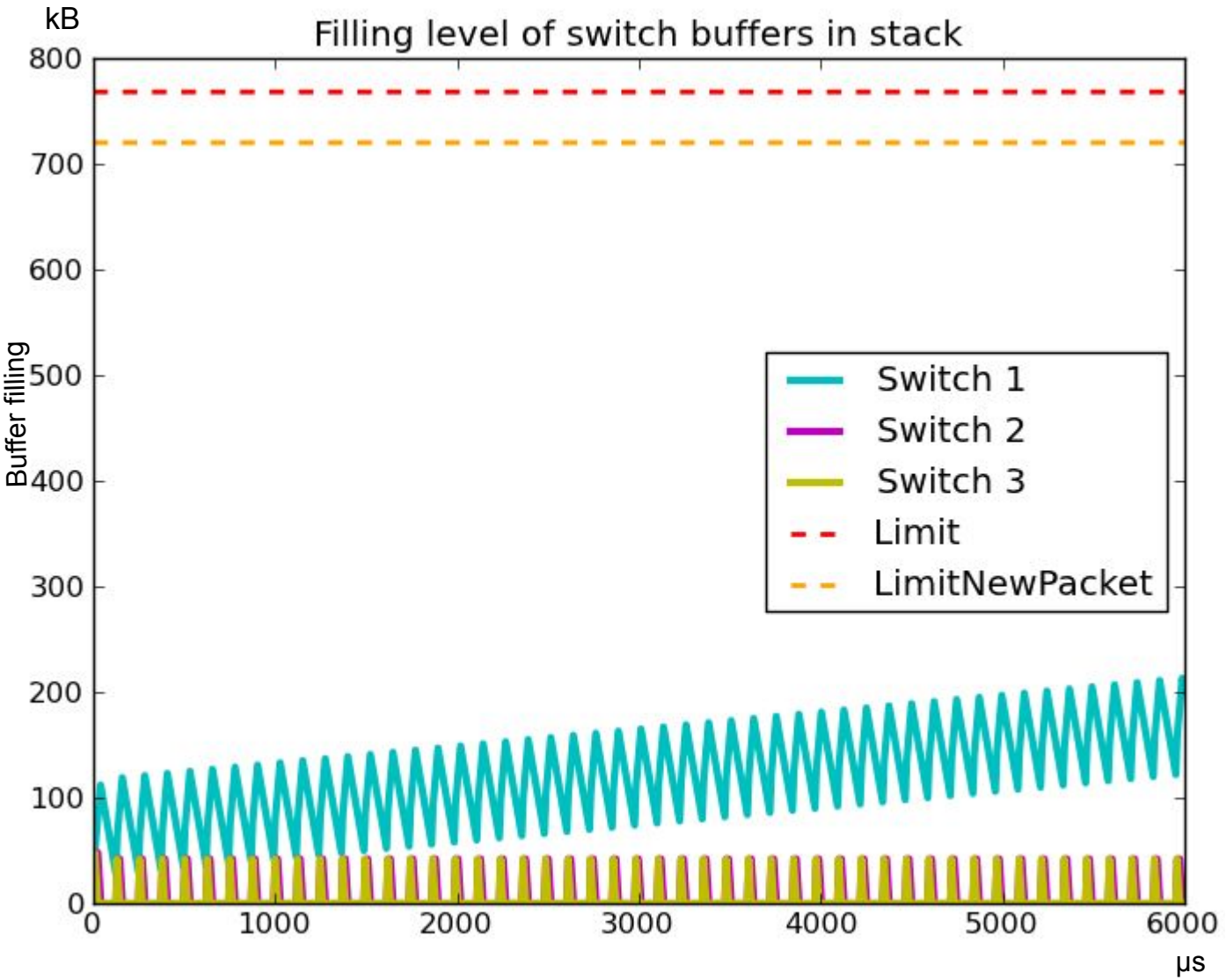


50 consecutive incoming events within 6250 µs
1 event each 125 µs

Switches stack model

Each switch has a capacity of **768 kB**

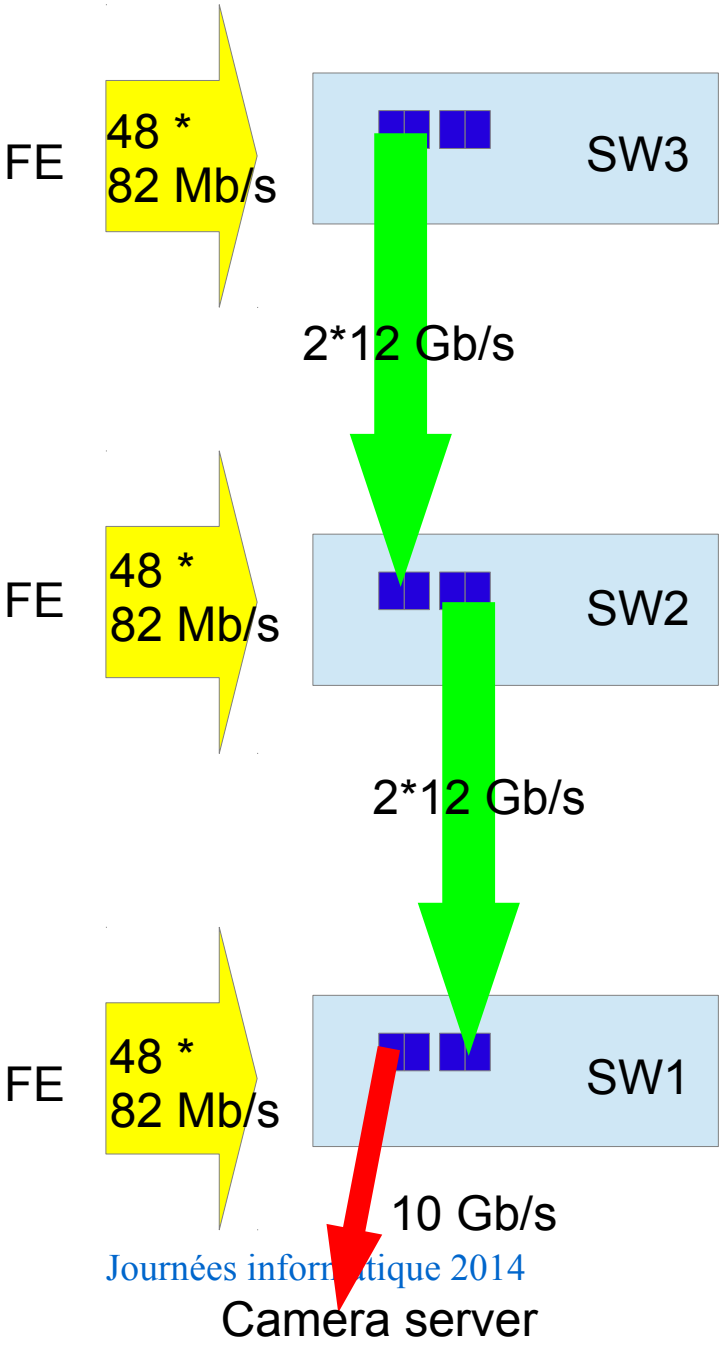


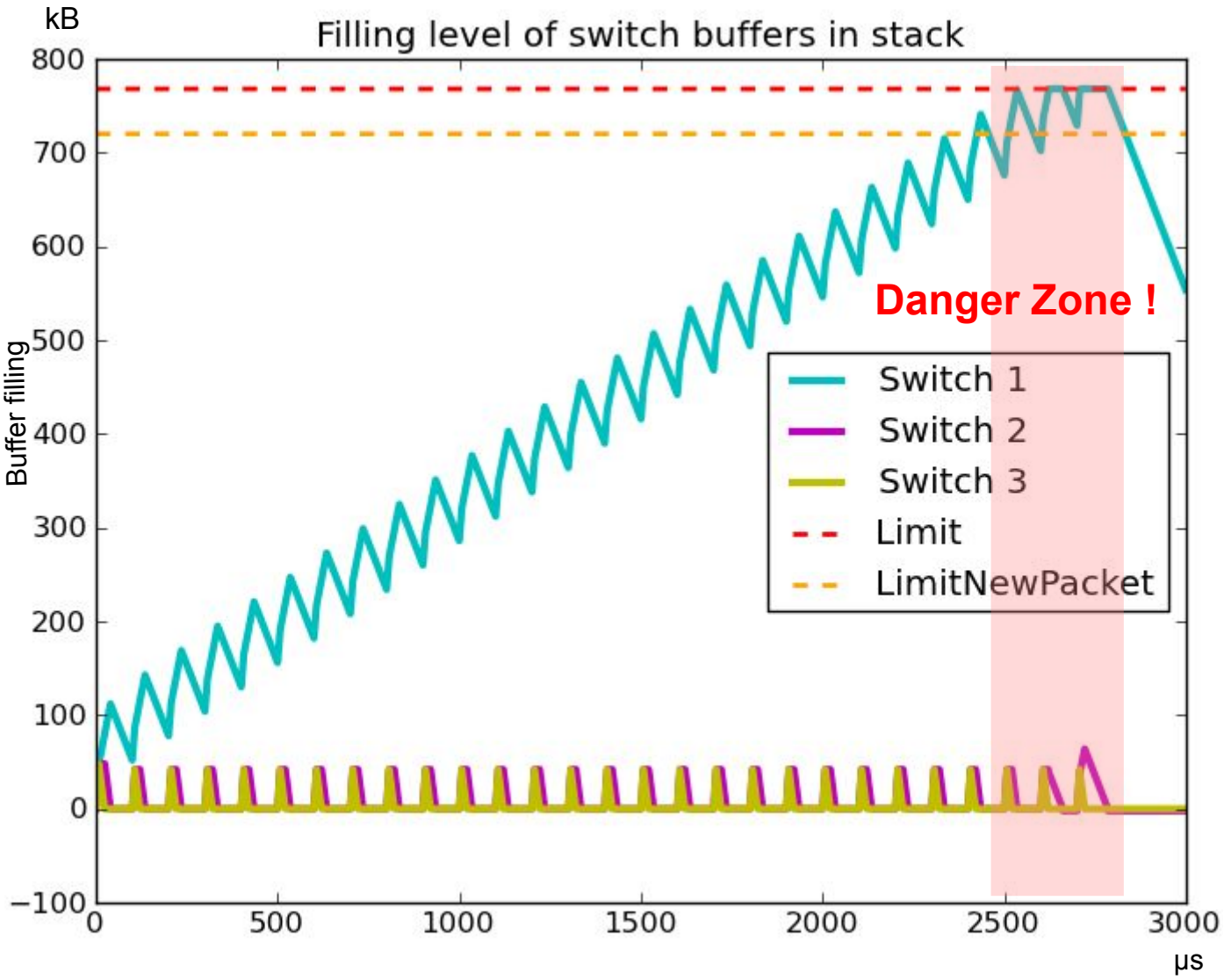


50 consecutive incoming events within 5800 µs
1 event each 116 µs

Switches stack model

Each switch has a capacity of **768 kB**

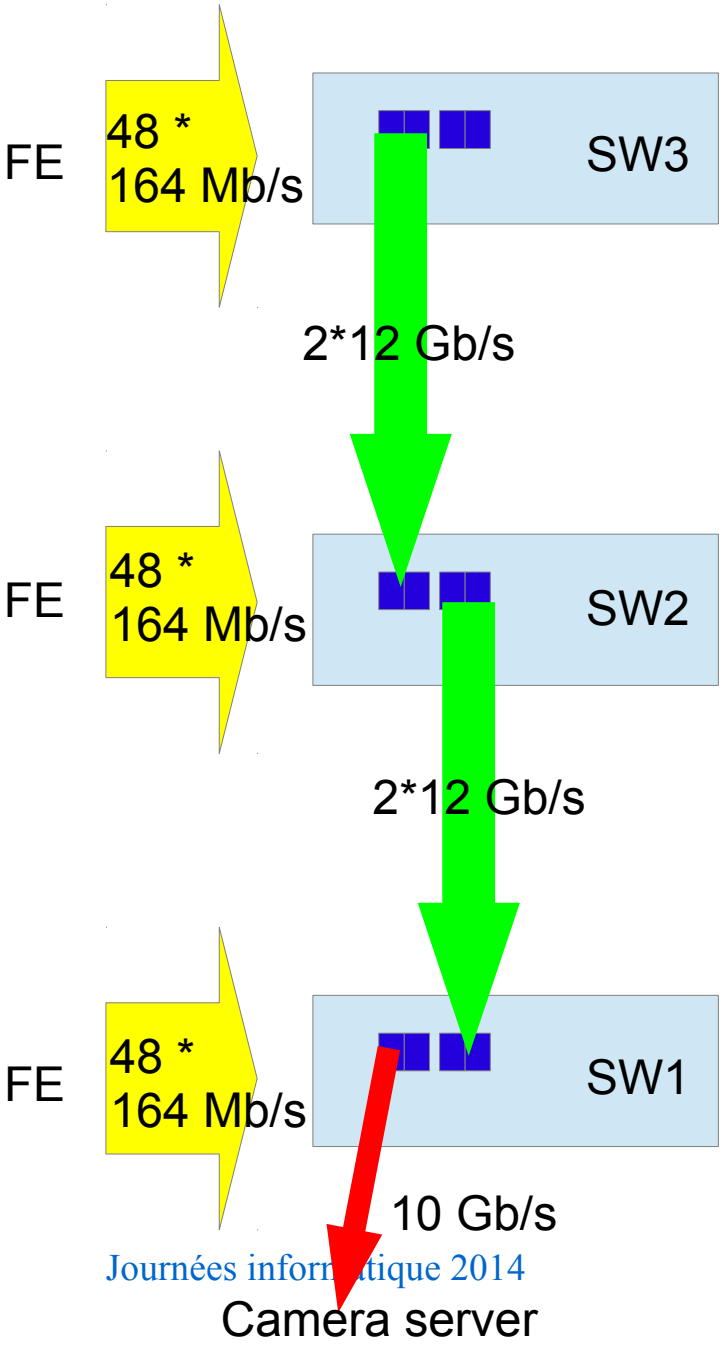


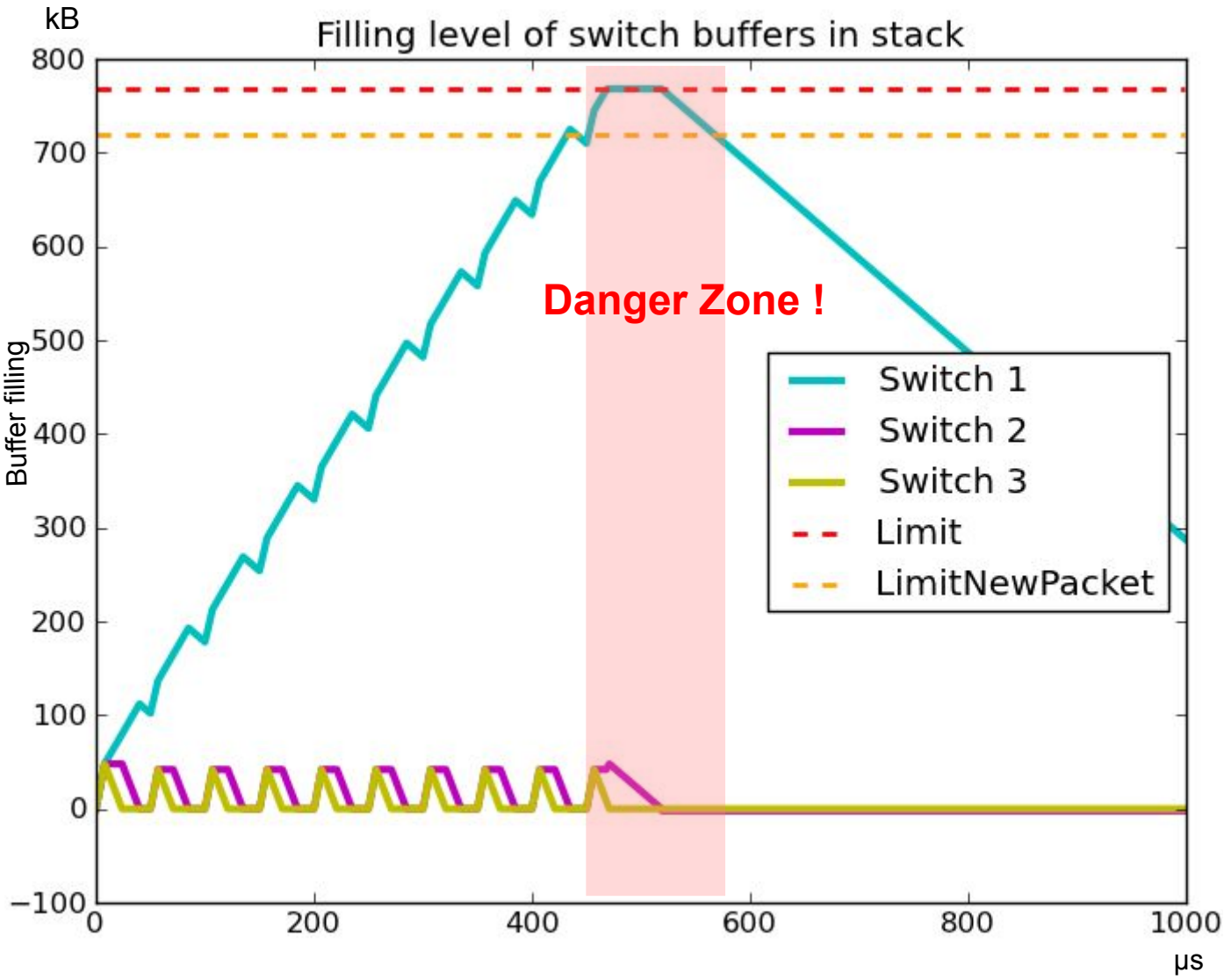


28 consecutive incoming events within 2800 µs
1 event each 100 µs

Switches stack model

Each switch has a capacity of **768 kB**

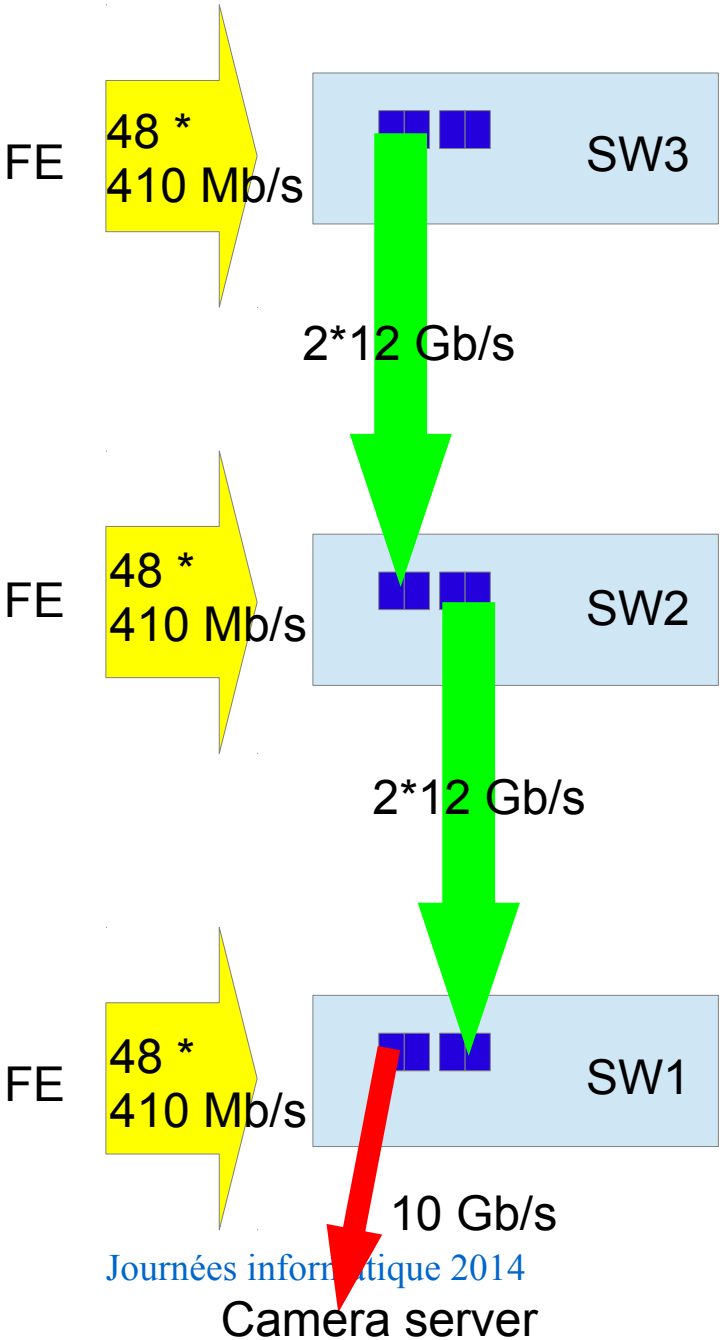


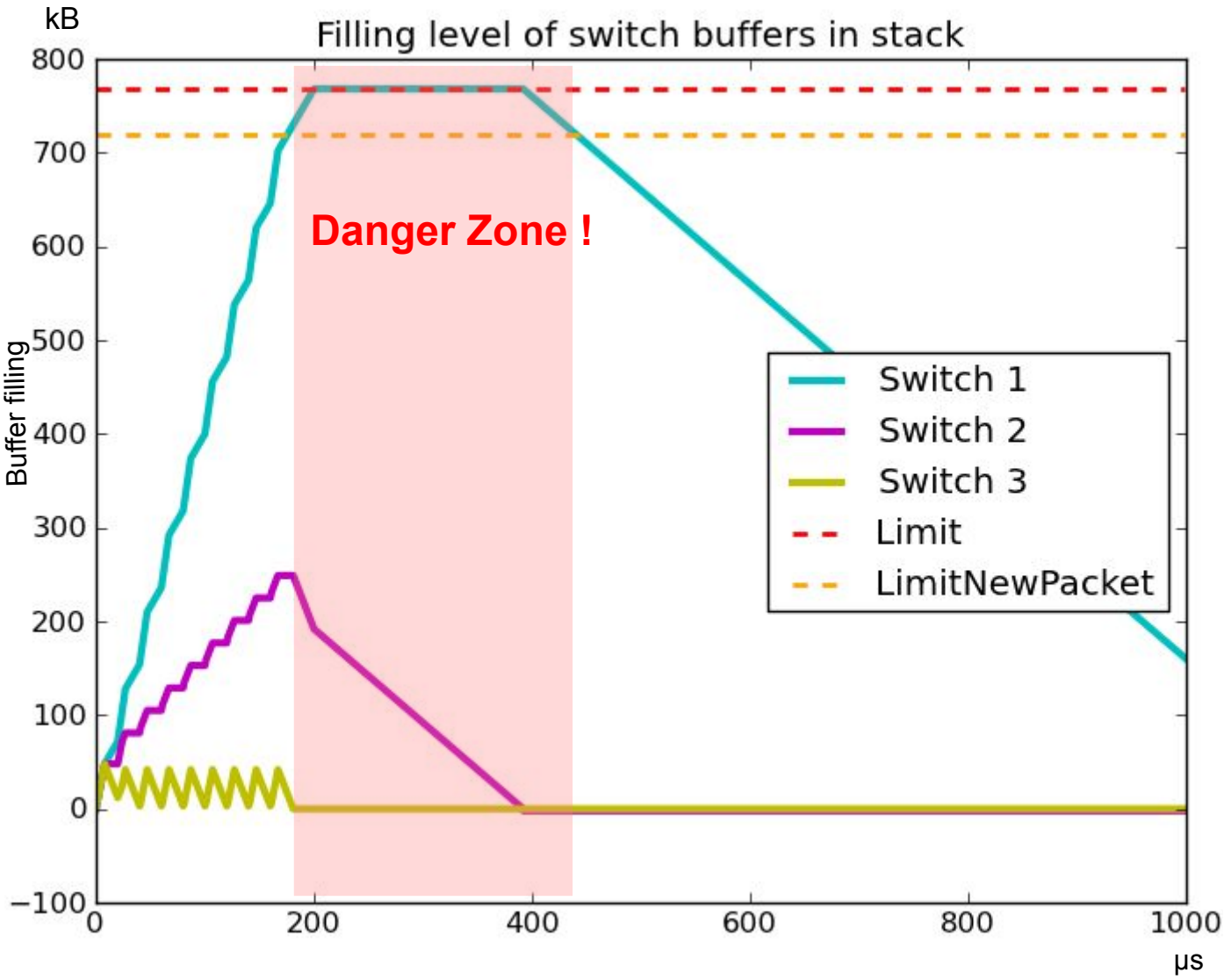


10 consecutive incoming events within 500 µs
1 event each 50 µs

Switches stack model

Each switch has a capacity of **768 kB**

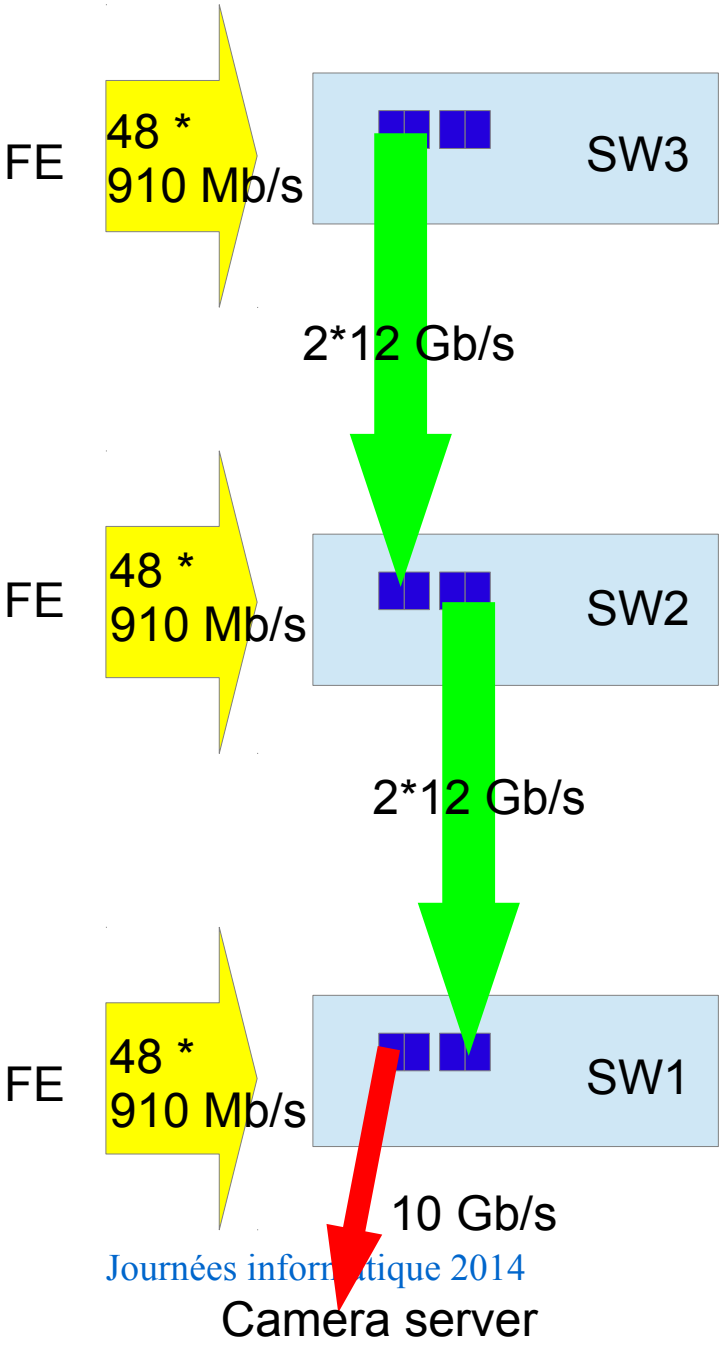


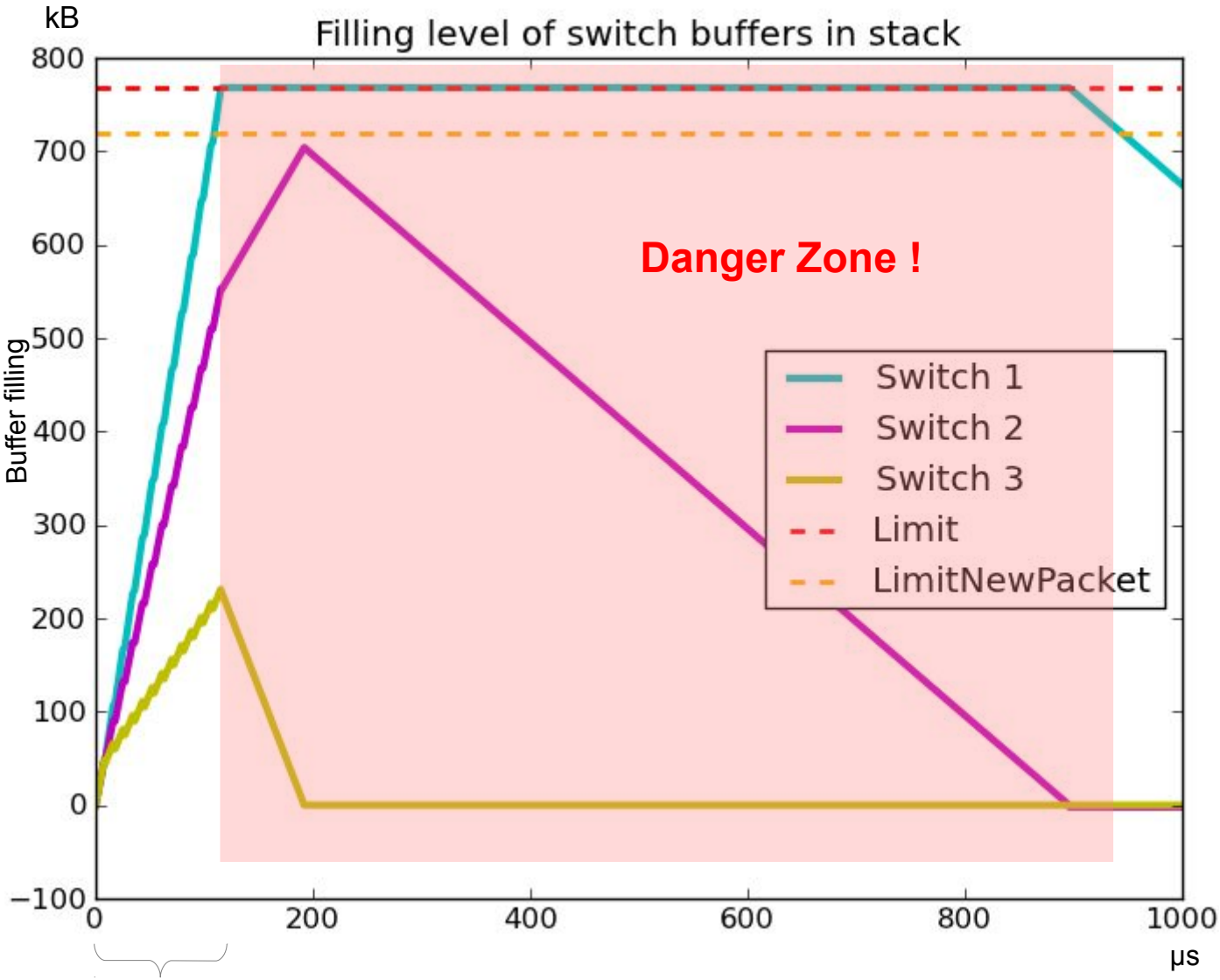


9 consecutive incoming events within 180 µs
1 event each 20 µs

Switches stack model

Each switch has a capacity of **768 kB**

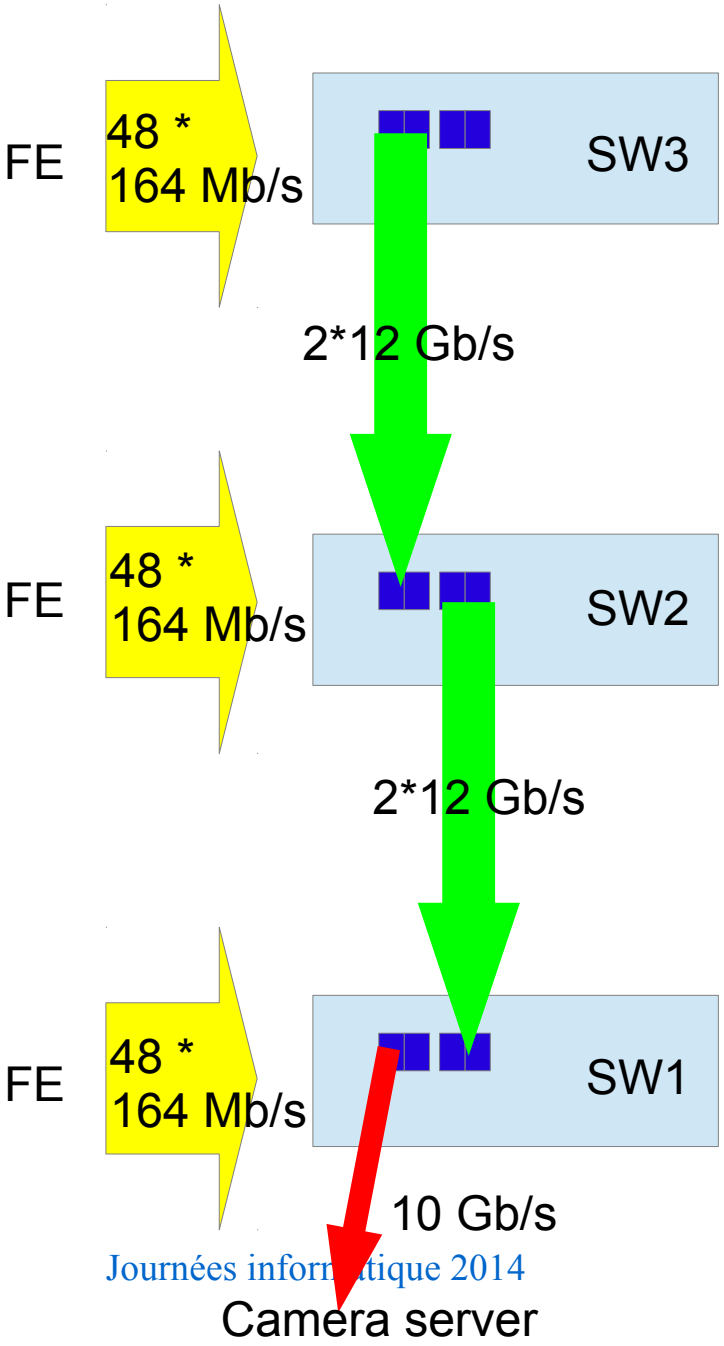


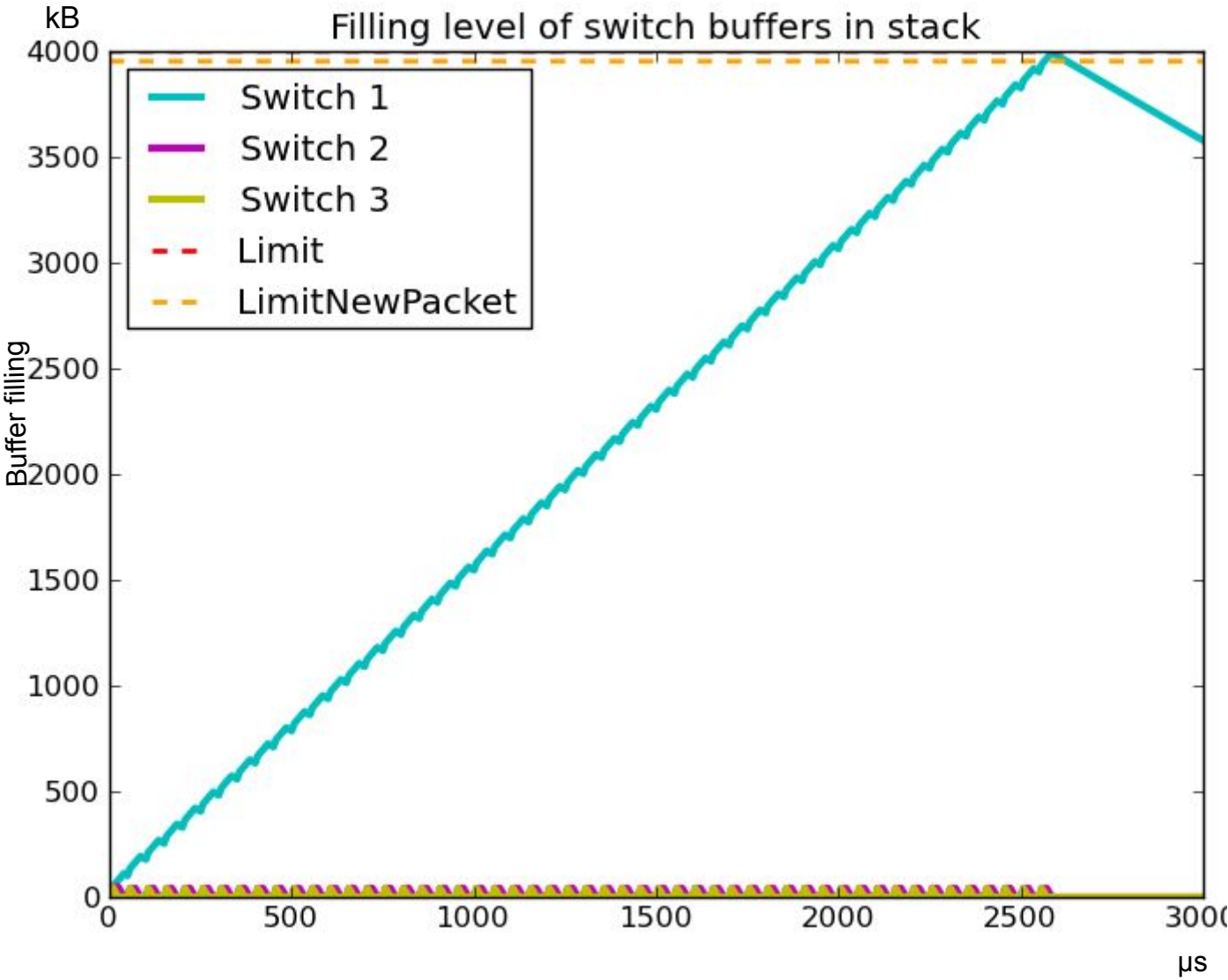


13 consecutive incoming events within 117 µs
1 event each 9 µs

Switches stack model

Each switch has a capacity of **4000 kB**

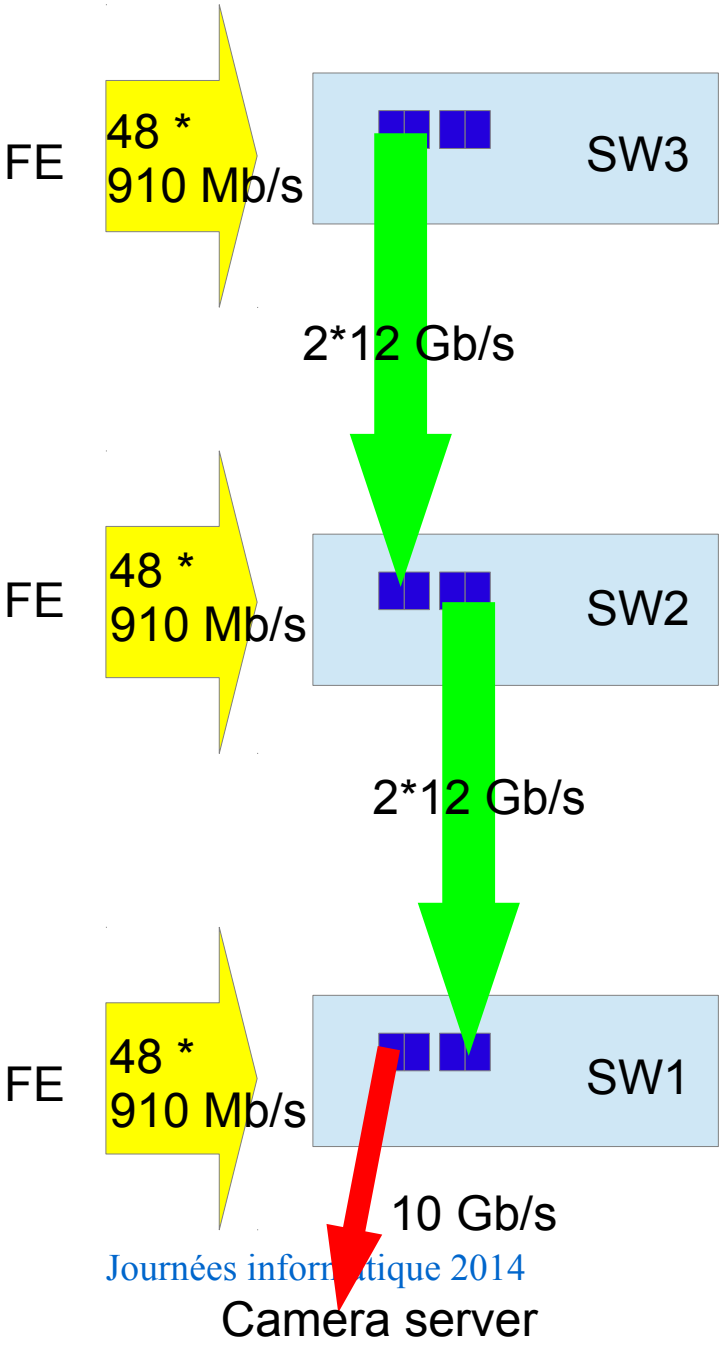


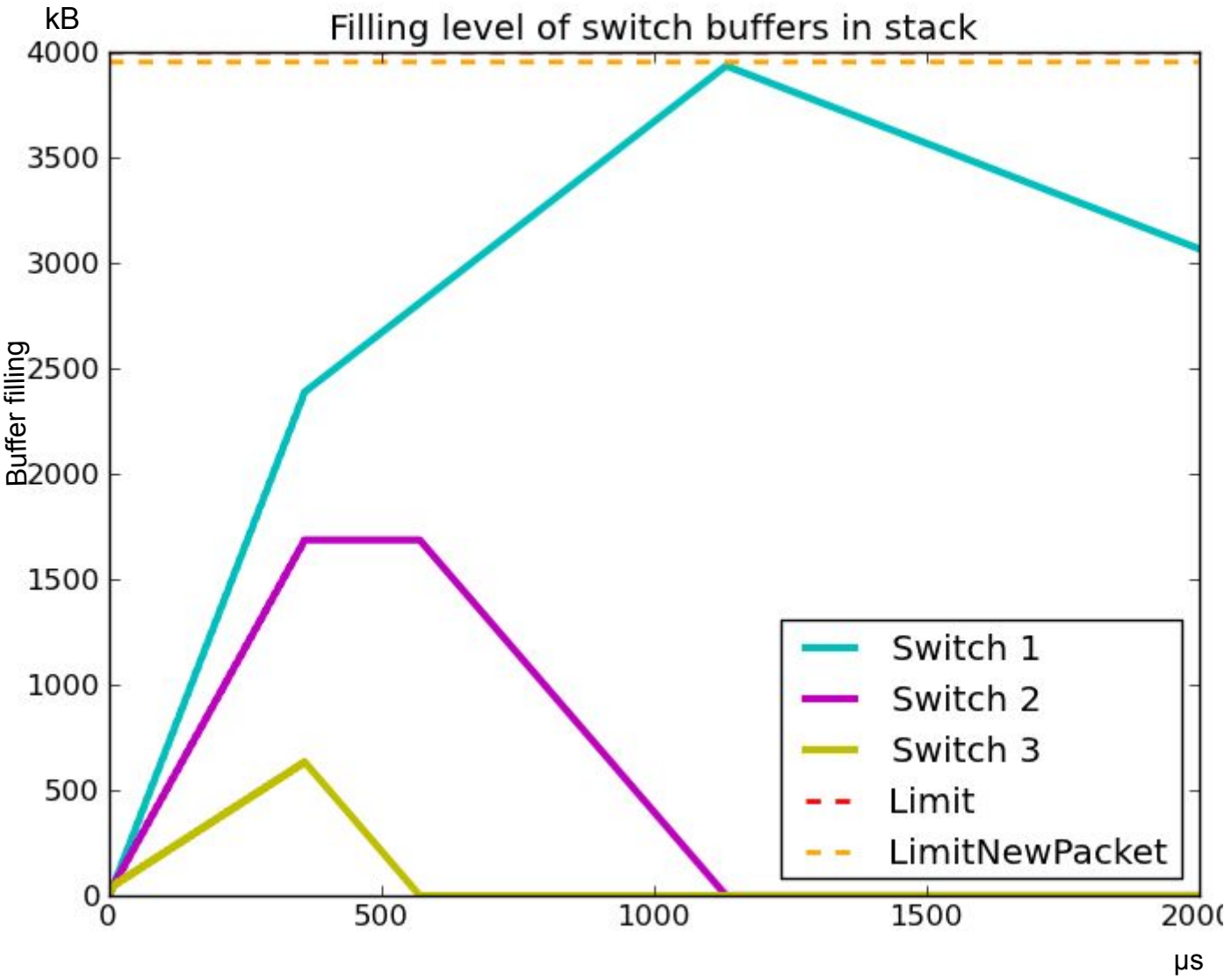


52 consecutive incoming events within 2600 µs
1 event each 50 µs

Switches stack model

Each switch has a capacity of **4000 kB**





40 consecutive incoming events within 360 µs
1 event each 9 µs

Switches examples

Dell PowerConnect 6248 Public price : 2600 Euros -> **15600 euros/Telescope**

BO = Buffer size = **768 kB**

Dell PowerConnect 7048 Public price : 3700 Euros -> **22200 euros/Telescope**

BO = Buffer size = **4 MB**

Problem :
Commercial switches don't seem to be well suited to our data flow model
(from big producers to single consumers)

According to the Dell engineers, the **packets storage buffer** is not common to all ports but **equally assigned to each port** (even the stacking and 10Gb ports !!??) and no flow control mechanism present on stacking links

⇒ High rates ports buffers very **quickly saturated** (with a few events)

Switches examples

Dell Force 10 S60

Public price : 9200 Euros -> **55200 euros/Telescope**

BO = Buffer size = **1.25 GB**

The operating system of this switch allows **custom memory allocation**

⇒ With bigger buffers for the higher rates ports, **saturation should occur later**

Switches examples

Dell Force 10 S60

Public price : 9200 Euros -> **55200 euros/Telescope**

BO = Buffer size = **1.25 GB**

The operating system of this switch allows **custom memory allocation**

⟹ With bigger buffers for the higher rates ports, **saturation should occur later**

We will elaborate more sophisticated models

Premiers tests

Tests de synchronicité

Yassir Moudden (CEA Saclay) a développé un générateur de paquets sur 48 ports synchronisés à la nanoseconde.
Les premiers tests montrent une perte de paquets certainement due à un récepteur pas assez fiable.

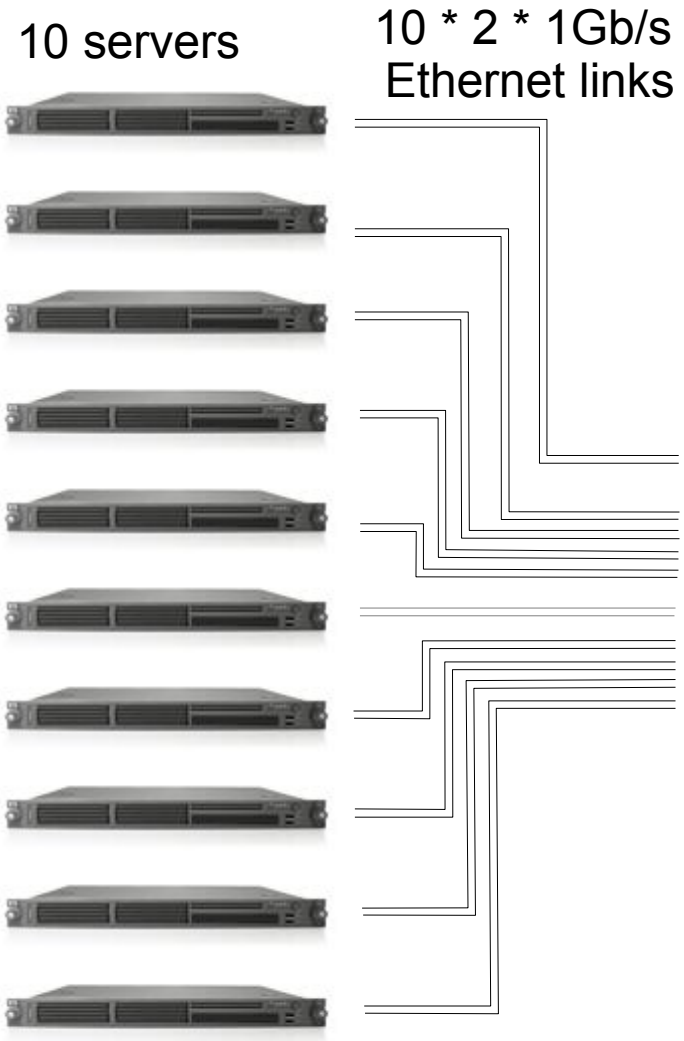
La reproduction au CPPM avec le stimulateur montre les mêmes limites du récepteur.
Par contre, en utilisant une carte Napatech NT4E en réception sur un port Gigabit, aucune perte de paquets jusqu'à 40 paquets consécutifs (à 1 Gb/s) sur 42 ports.

Tests de profondeur de buffer

Des tests de profondeur de buffer vont être réalisés avec des scénarios aléatoires lancés sur le stimulateur.

Camera server software prototype

Test configuration

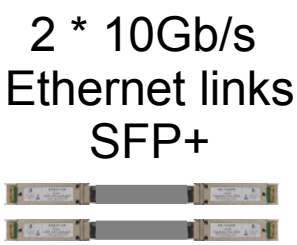


Powerconnect 6248

Or

Force10 S60

2 * 10Gb/s
Ethernet links
SFP+



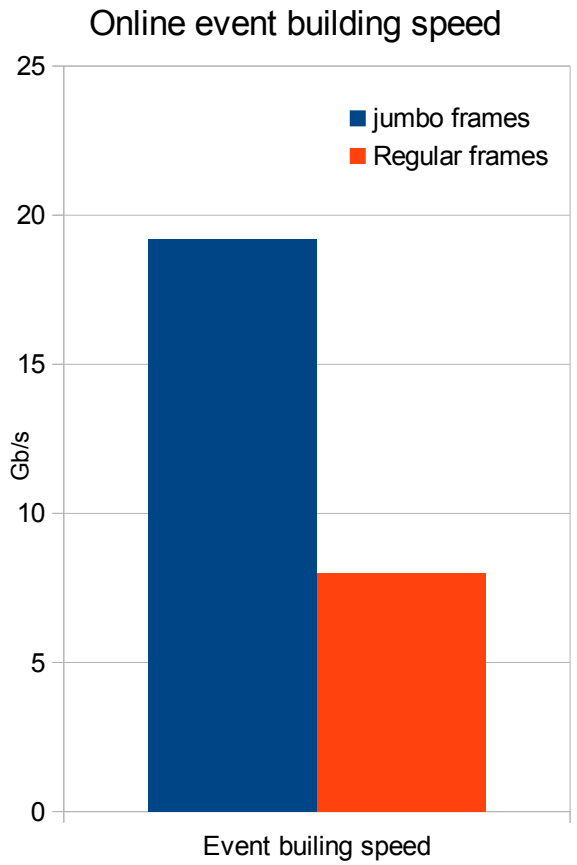
Dell T7500 workstation

Event building measurements

Online Event building speed	
Jumbo frames (8192 bytes) : 19.2 Gb/s (2.4 GB/s) with no loss Average CPU usage : 160 % (1.6 cores/12) ~ 8000 events/s built	Regular frames (1024 bytes) : 8 Gb/s (1 GB/s) with no loss Average CPU usage : 170 % (1.7 cores/12) ~ 3300 events/s built

- Results obtained with standard libraries
- 300 stimulation nodes generate data
- Incoming data through two 10 Gb adapters
- Two « channels » used
- CPU load spread on 2 cores (on the same multiprocessor)

The regular frames issue

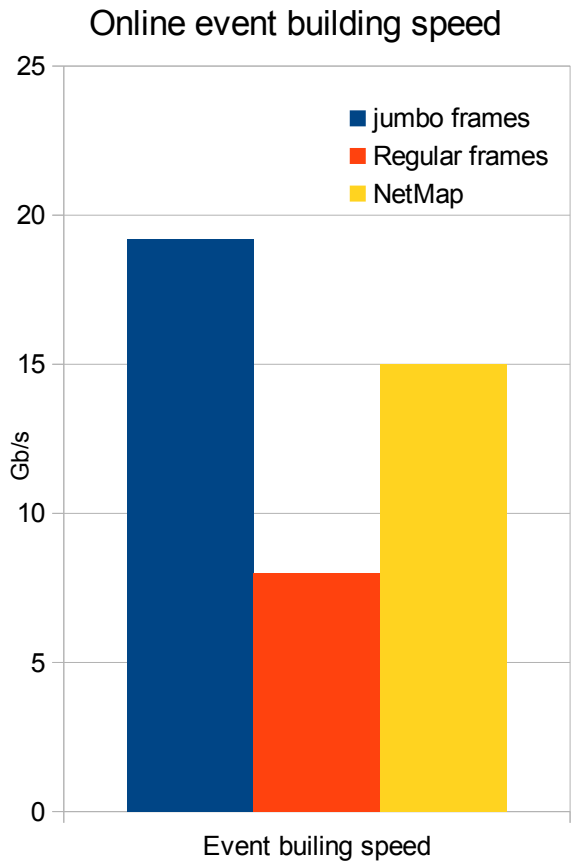


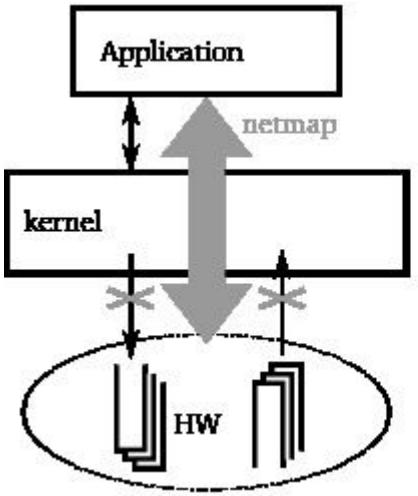
- Better results with jumbo frames
- But must be compatible with electronics

⇒ Must improve smaller packets reception

- Replace inefficient network software achitectures by direct access to network components

The regular frames issue : Netmap





- We reach an event building at 15 Gb/s with 1024 bytes packets
- CPU usage ~ 170 %

Netmap : a novel framework for fast packet I/O
Luigi Rizzo, Università di Pisa, Italy
Proceedings of the 2012 USENIX Annual Technical Conference, June 2012

Last results for data reception only

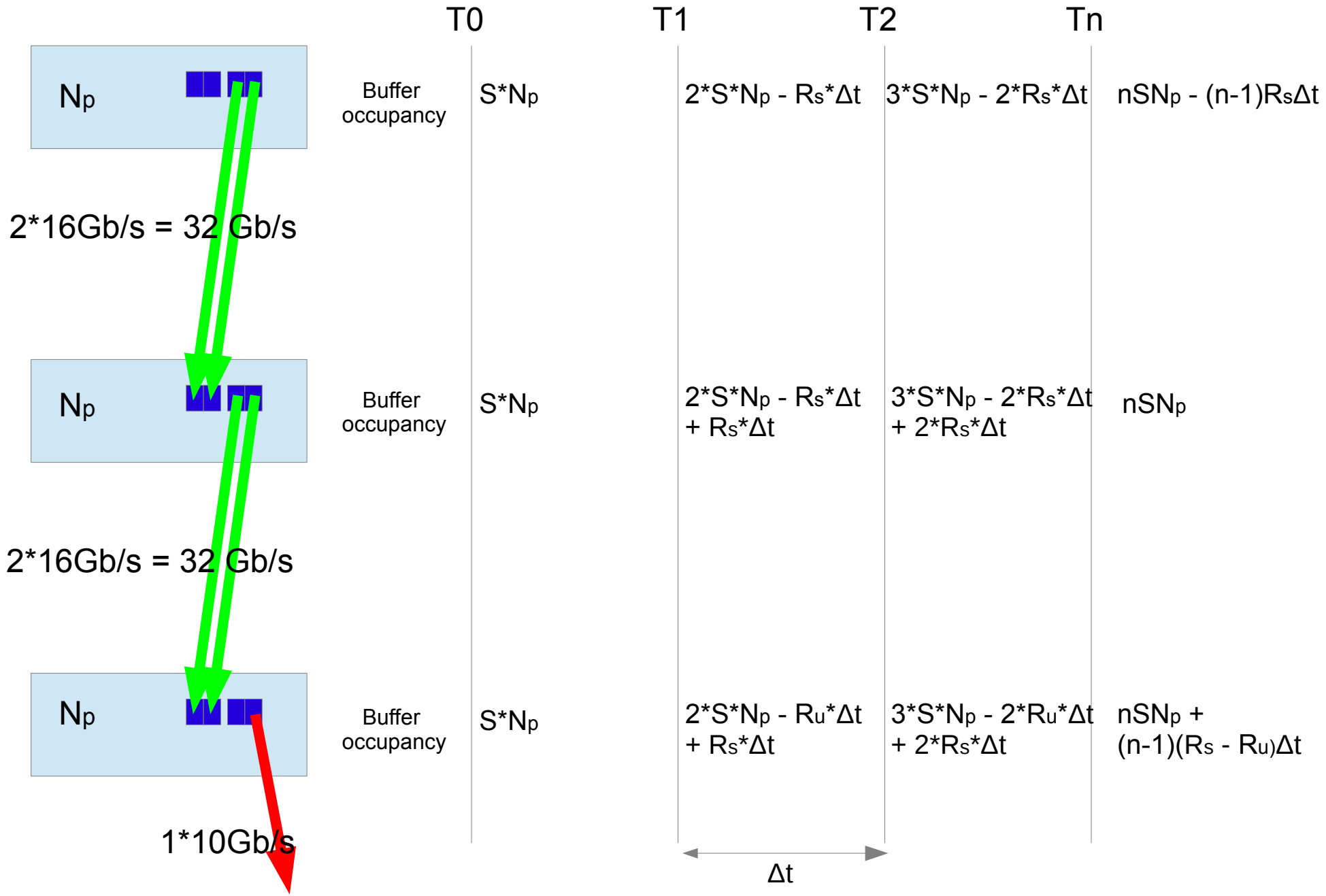
Last results with Netmap and stimulator:

Data reception at 19.2 Gb/s with small data loss (less than 0.03 %)

On two tasks (1 per 10Gb port) with isolated CPU, NUMA aware allocation...

→ We should avoid data loss by distributing packets reception on several tasks (CPUs) with netmap and RSS

Backup


T0
T1
T2
Tn
Np
Buffer occupancy
S*Np
2*S*Np - Rs*Δt
3*S*Np - 2*Rs*Δt
nSNp - (n-1)RsΔt
2*16Gb/s = 32 Gb/s
Np
Buffer occupancy
S*Np
2*S*Np - Rs*Δt + Rs*Δt
3*S*Np - 2*Rs*Δt + 2*Rs*Δt
nSNp
2*16Gb/s = 32 Gb/s
Np
Buffer occupancy
S*Np
2*S*Np - Ru*Δt + Rs*Δt
3*S*Np - 2*Ru*Δt + 2*Rs*Δt
nSNp + (n-1)(Rs - Ru)Δt
1*10Gb/s
Δt

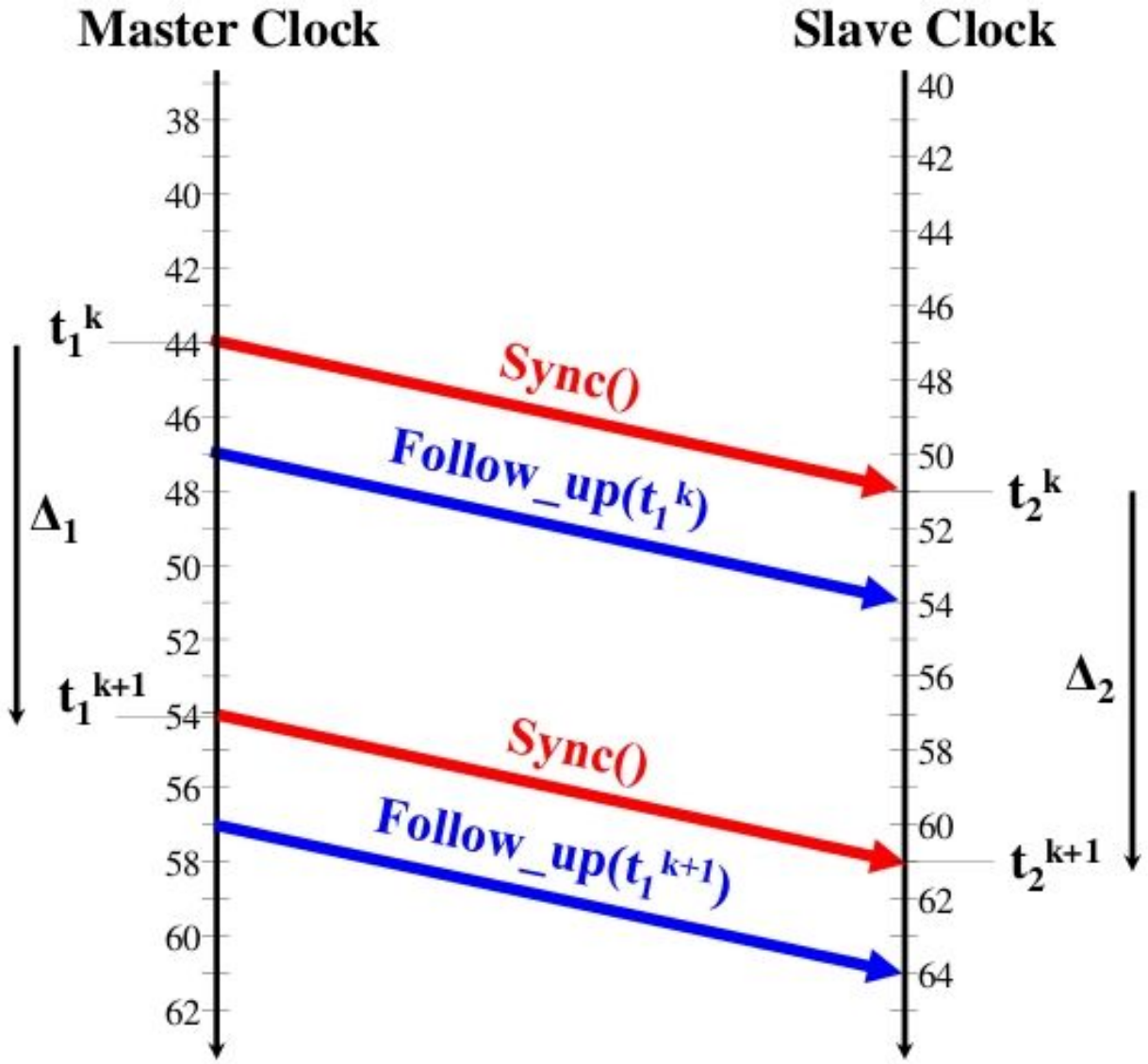
Stimulator : hardware time measurement

Napatech NT4E-4T

- 4 gigabit Ethernet ports
- Embedded memory for packets buffering
- Packets time stamping accuracy : 7 ns

Stimulator : Inter-boards synchronization

Clock drift determination



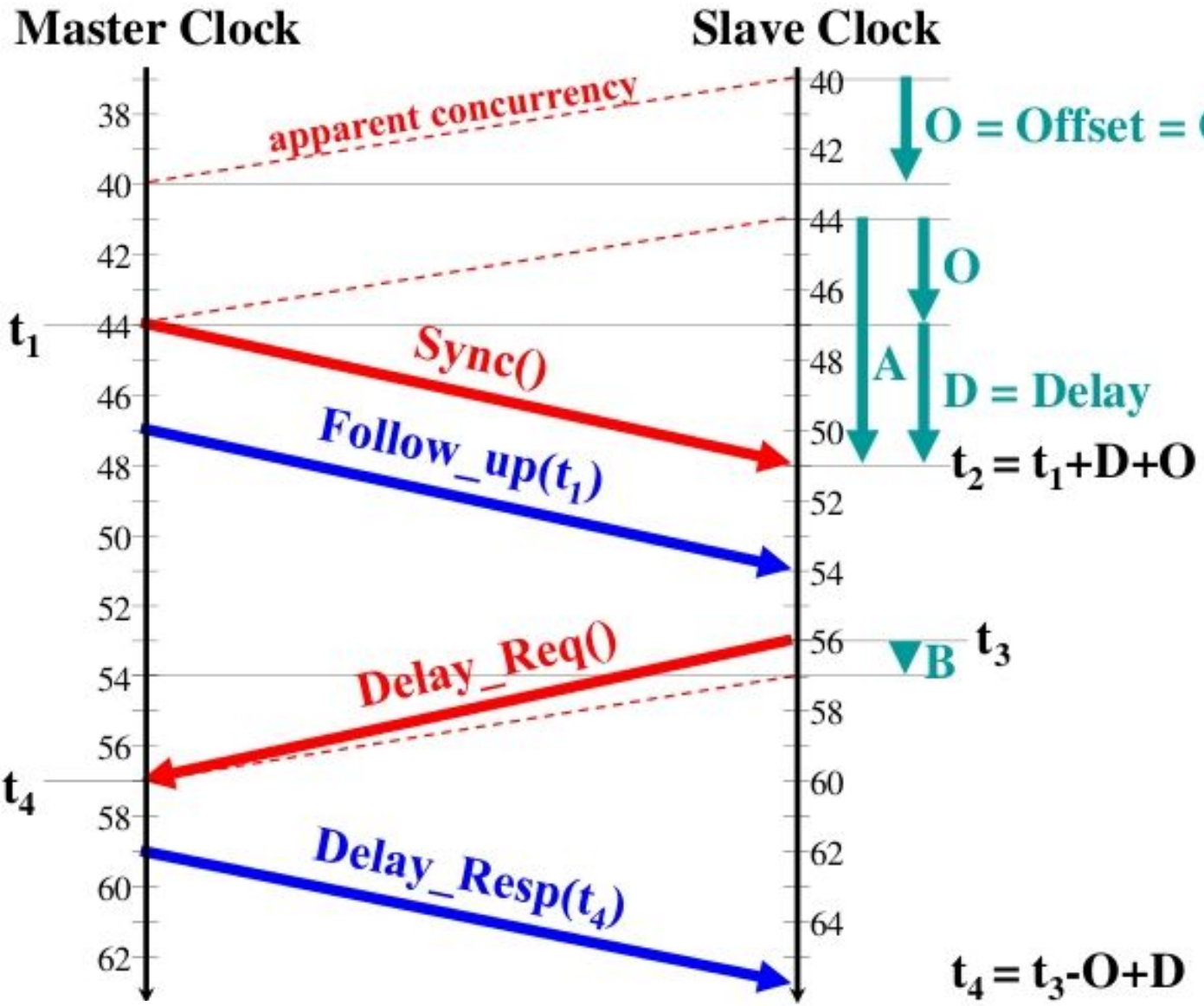
$$\Delta_1 = t_1^{k+1} - t_1^k$$

$$\Delta_2 = t_2^{k+1} - t_2^k$$

$$\text{Drift} = \frac{\Delta_2 - \Delta_1}{\Delta_2}$$

Stimulator : Inter-boards synchronization

Clock offset determination



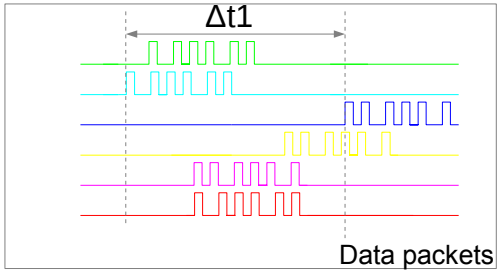
measured values t_1, t_2, t_3, t_4

$A = t_2-t_1 = D+O$

$B = t_4-t_3 = D-O$

$$\text{Delay } D = \frac{A+B}{2}$$

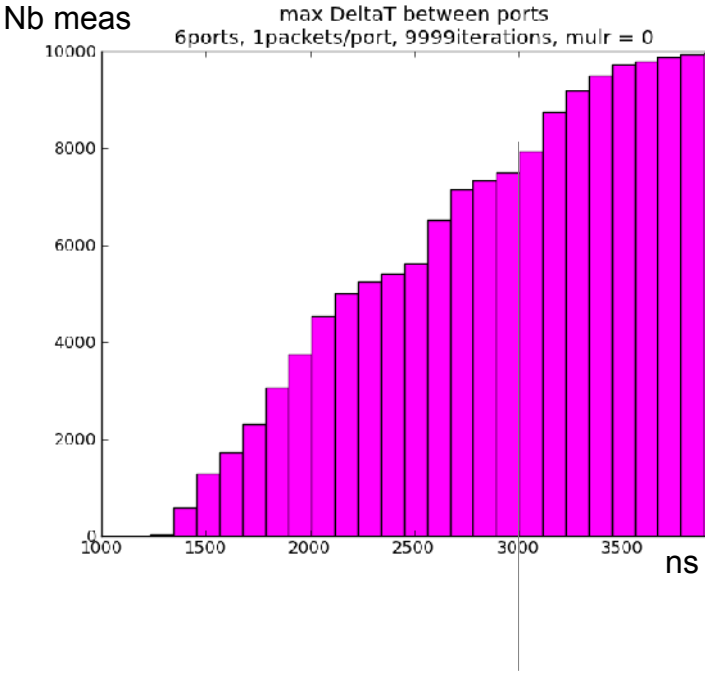
$$\text{Offset } O = \frac{A-B}{2}$$



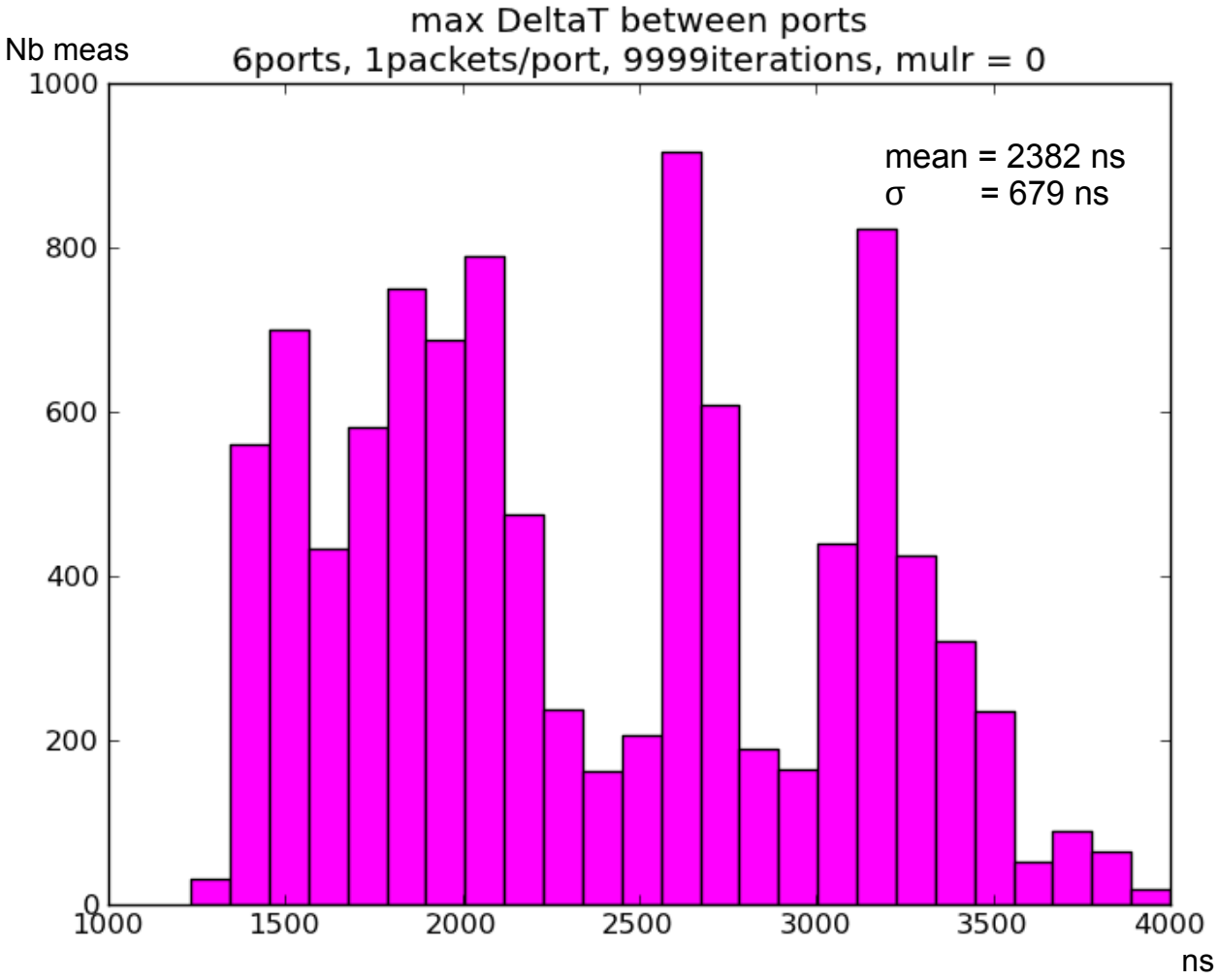
Stimulator : Board evaluation

Δt1 : time difference between the first packet sent and the last one on one SBC

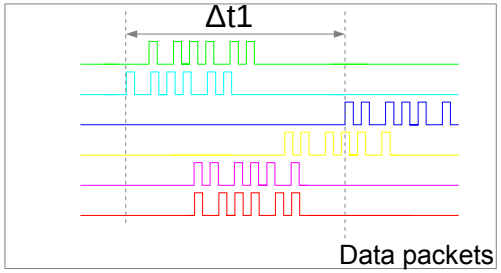
10000 measurements **6 ports** 1 packet per port



80 % < 3 µs



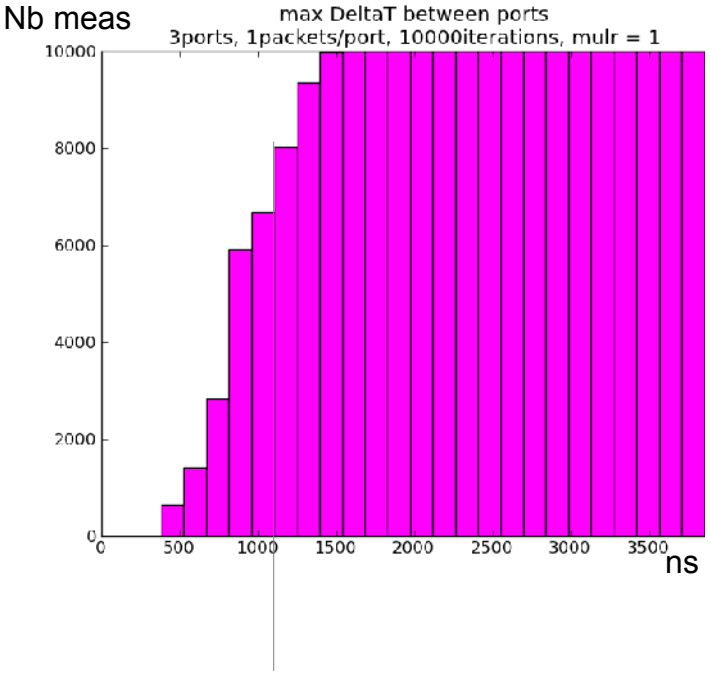
With EVOC Board,
Results slightly better with Jetway board



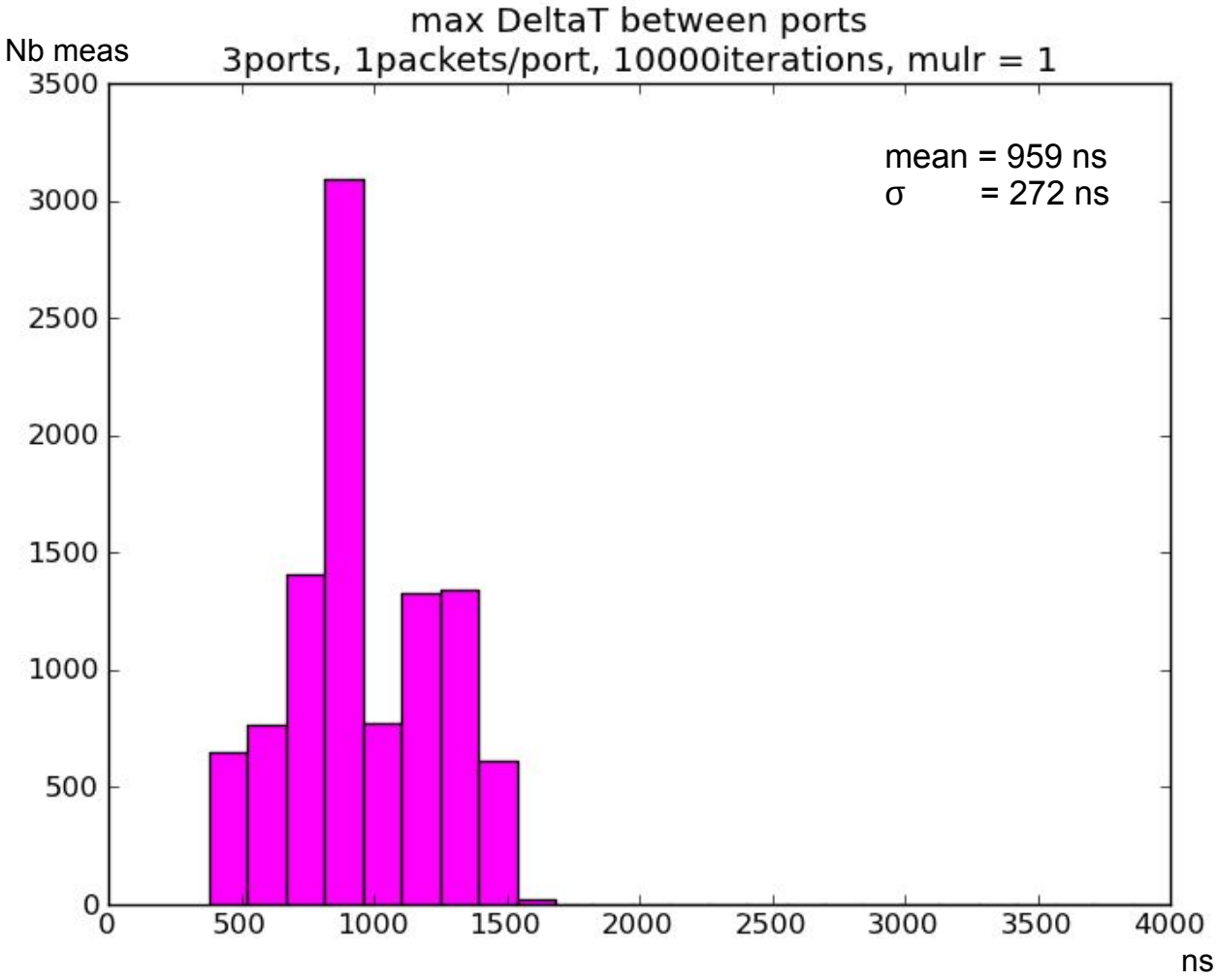
Stimulator : Board evaluation

Δt1 : time difference between the first packet sent and the last one on one SBC

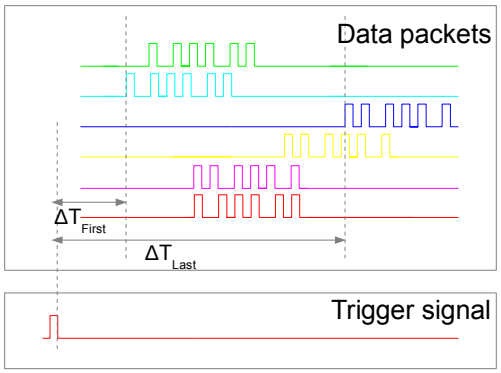
10000 measurements **3 ports** 1 packet per port



80 % < 1.2 µs



With EVOC Board,
Results slightly better with Jetway board



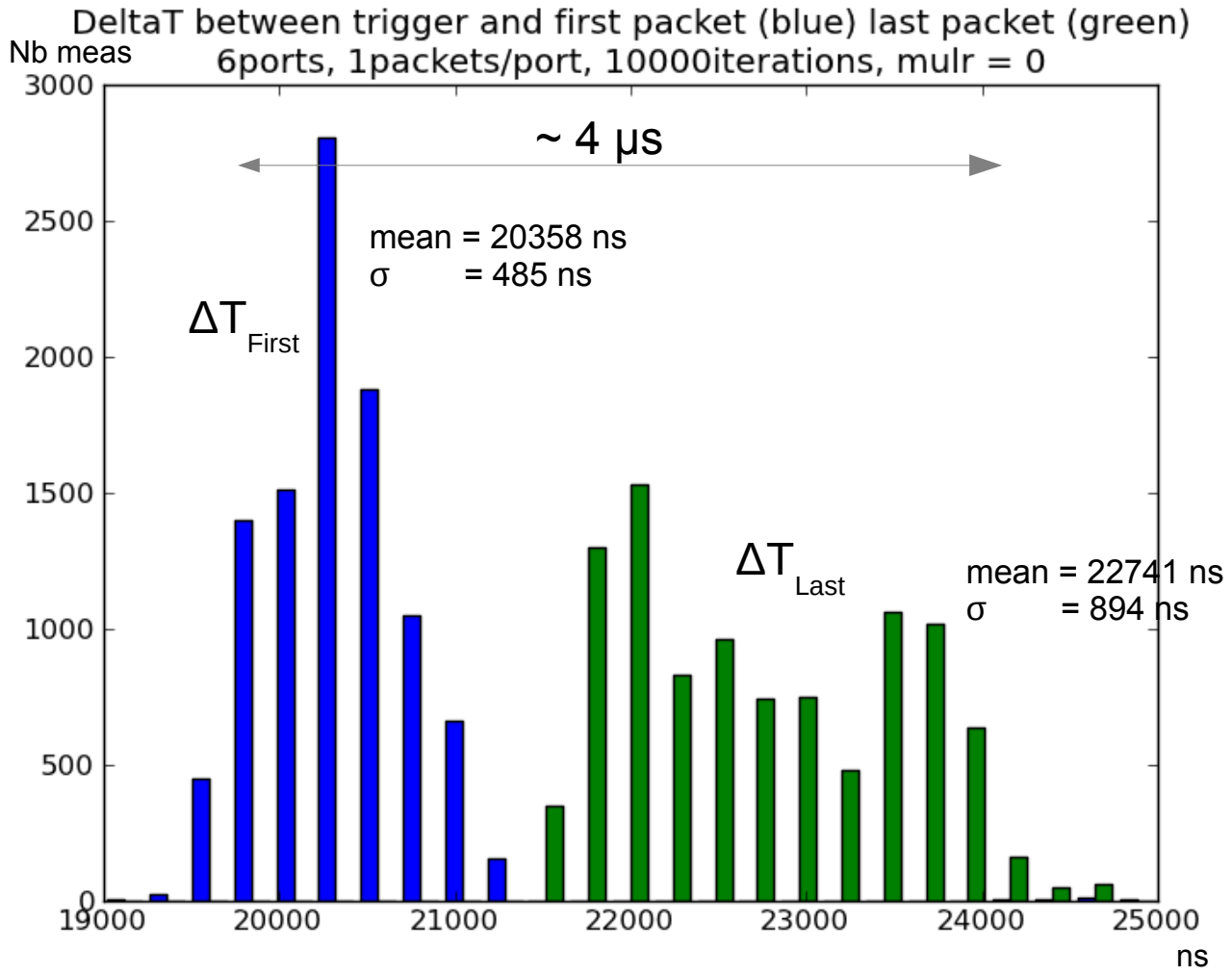
Stimulator : Board evaluation

ΔT_{First} : time difference between the send command and the first packet sent

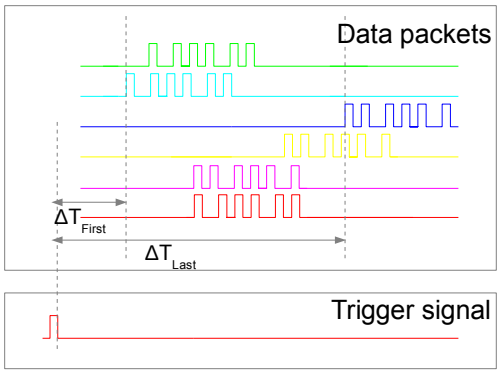
ΔT_{Last} : time difference between the send command and the last packet sent

10000 measurements **6 ports** 1 packet per port

Almost all paquets sent in a ~ 4 µs interval



With EVOC Board,
Results slightly better with Jetway board



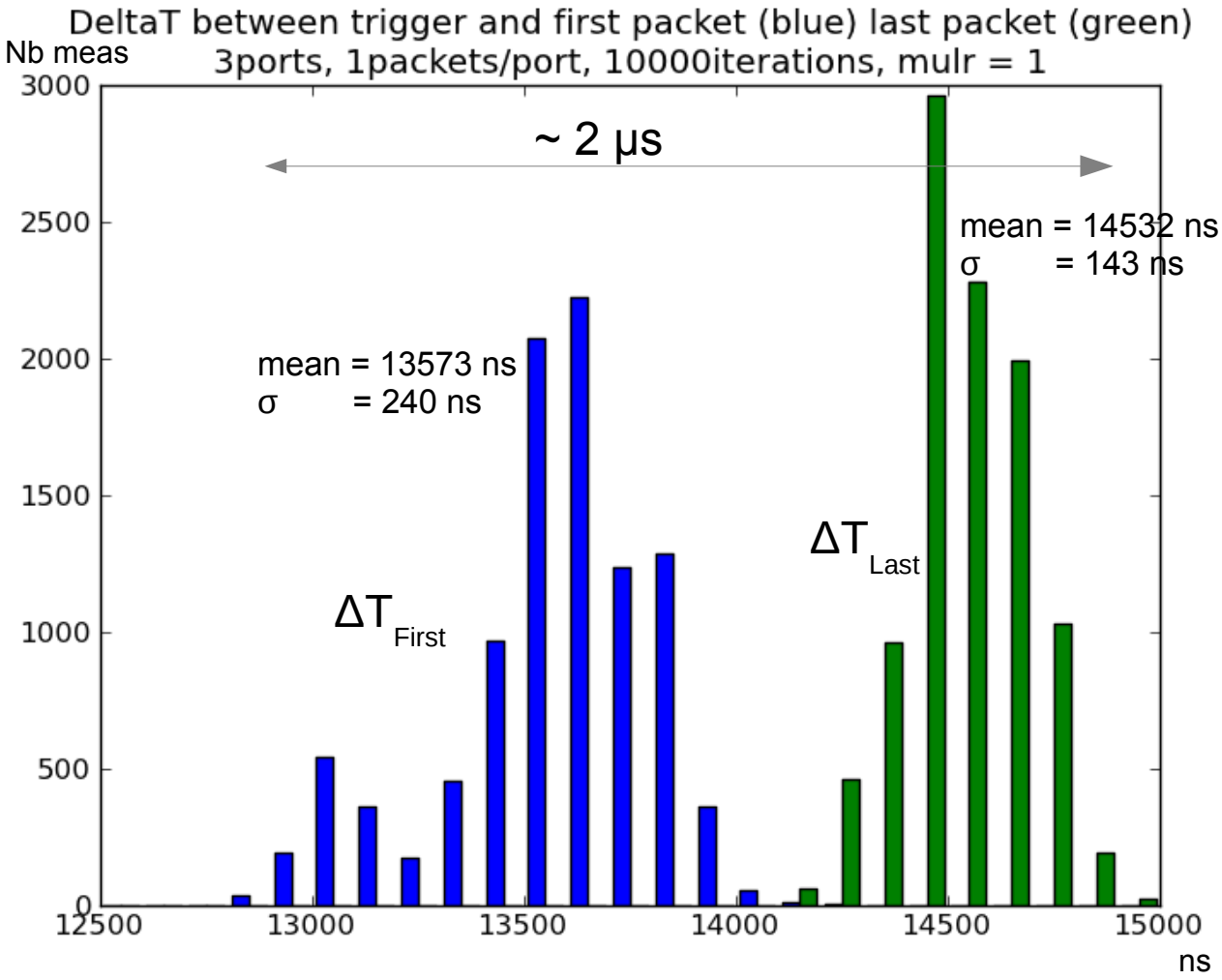
Stimulator : Board evaluation

ΔT_{First} : time difference between the send command and the first packet sent

ΔT_{Last} : time difference between the send command and the last packet sent

10000 measurements **3 ports** 1 packet per port

Almost all paquets sent in a ~ 2 µs interval



With EVOC Board,
Results slightly better with Jetway board

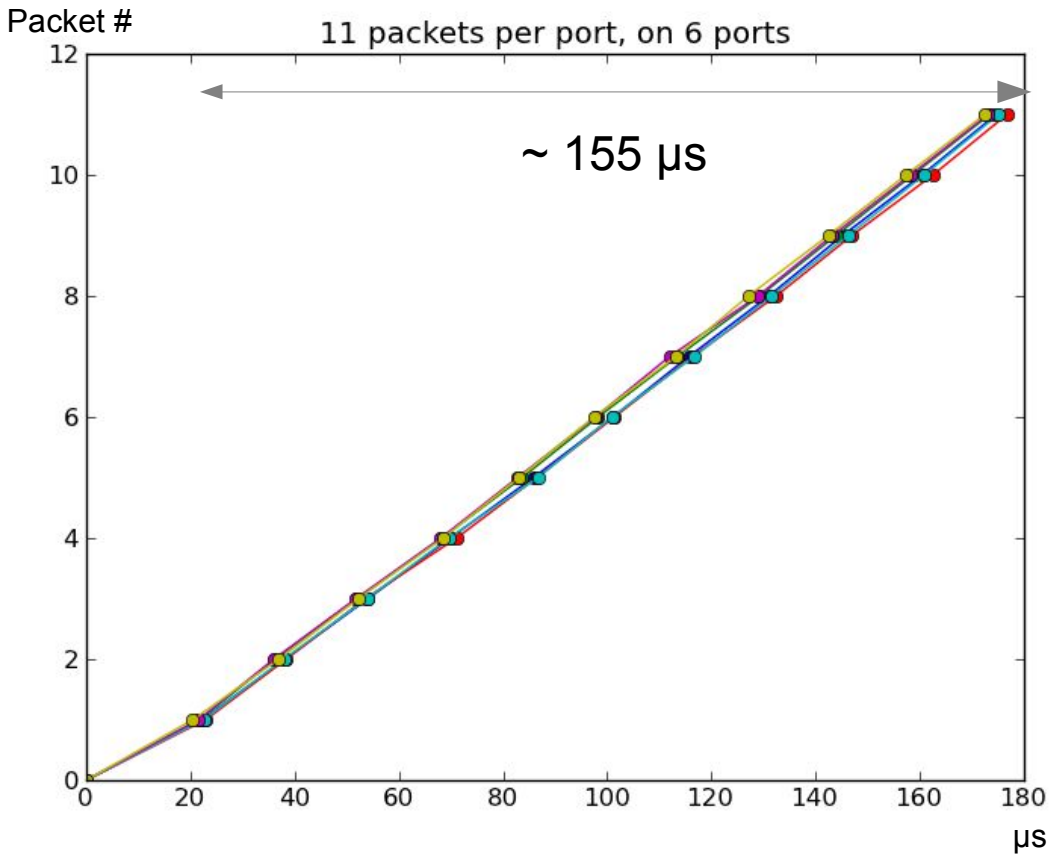
Stimulator : Board evaluation

Sending of 11 consecutive packets on each port

10000 measurements **6 ports** 11 packets per port

11 packets transferred in 155 µs

Data flow : 585 Mb/s

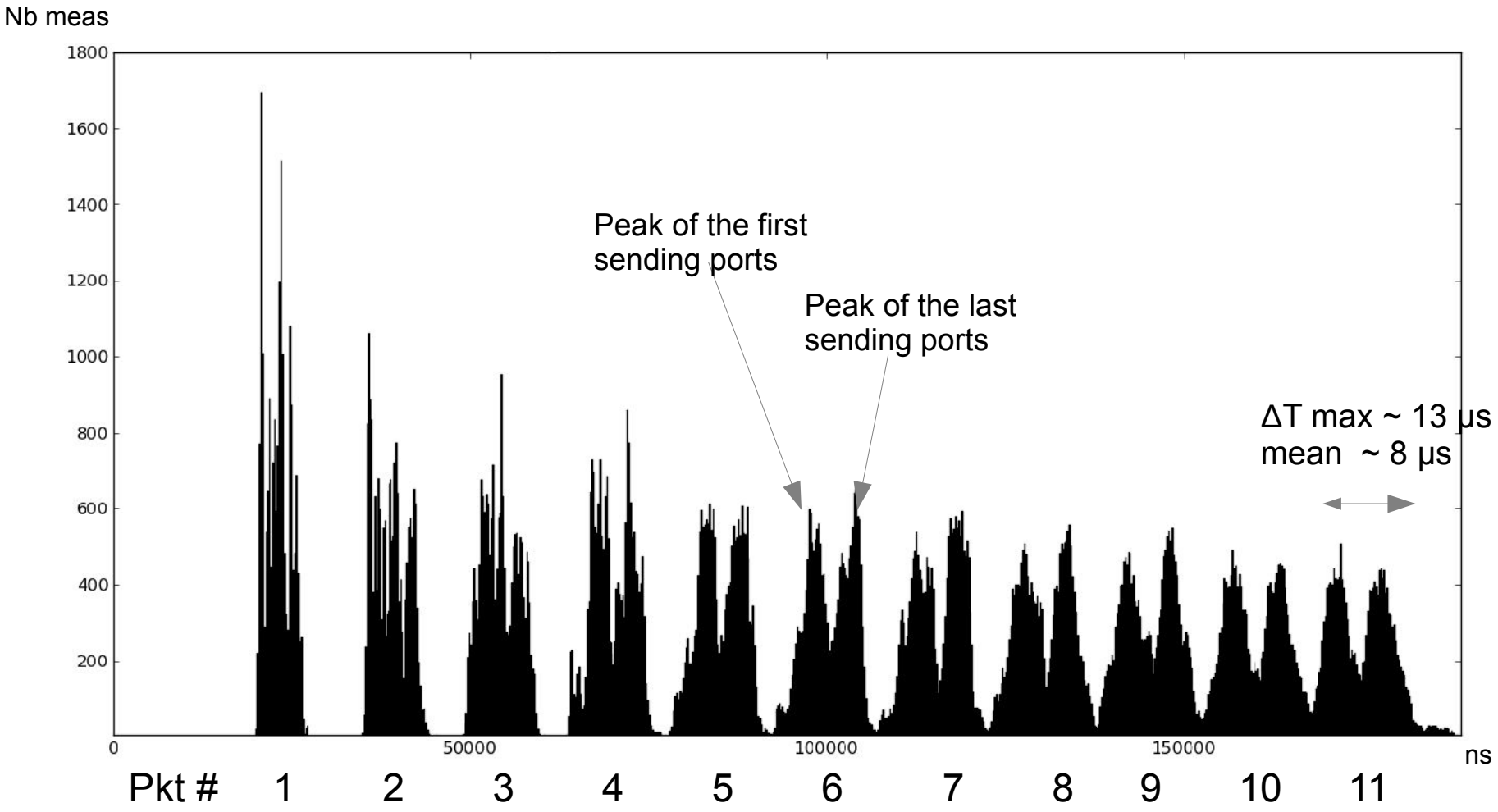


With EVOC Board,
Results slightly better with Jetway board

Stimulator : Board evaluation

Sending of 11 consecutive packets on each port :
First and last sending ports on one SBC

10000 measurements **6 ports** 11 packets per port



With EVOC Board,
Results slightly better with Jetway board

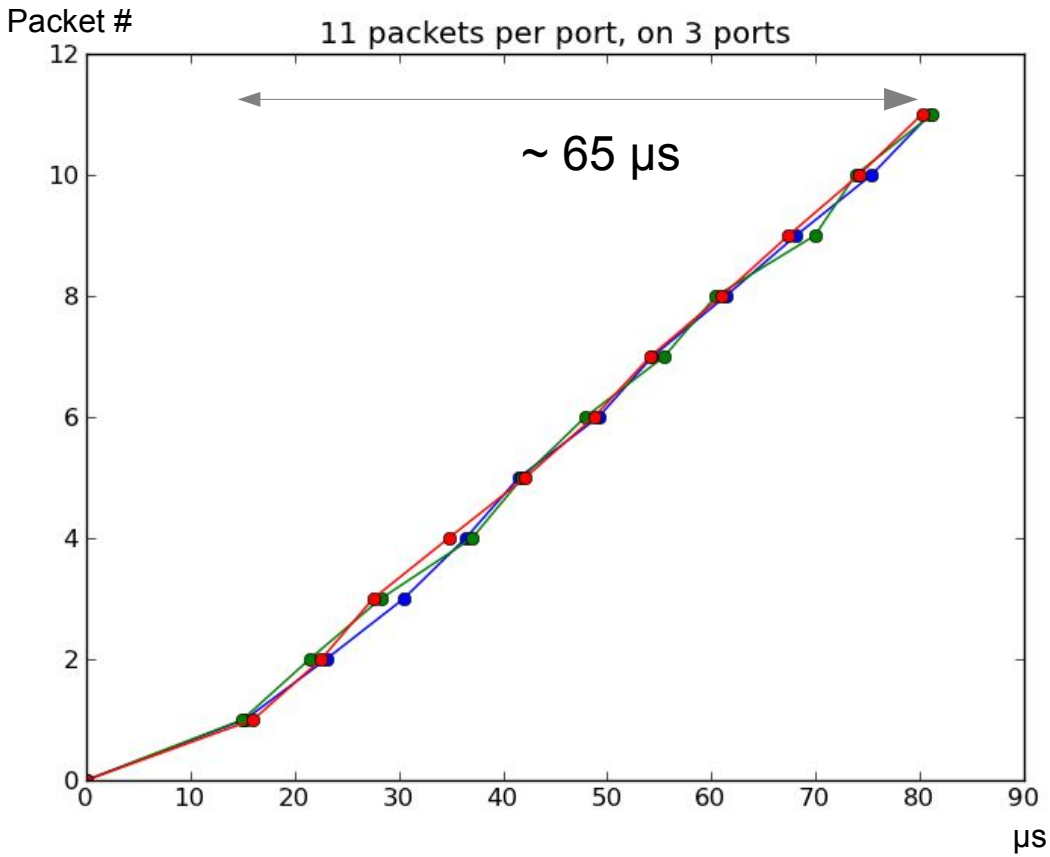
Stimulator : Board evaluation

Sending of 11 consecutive packets on each port

10000 measurements **3 ports** 11 packets per port

11 packets transferred in 65 µs

Data flow : 1.4 Gb/s

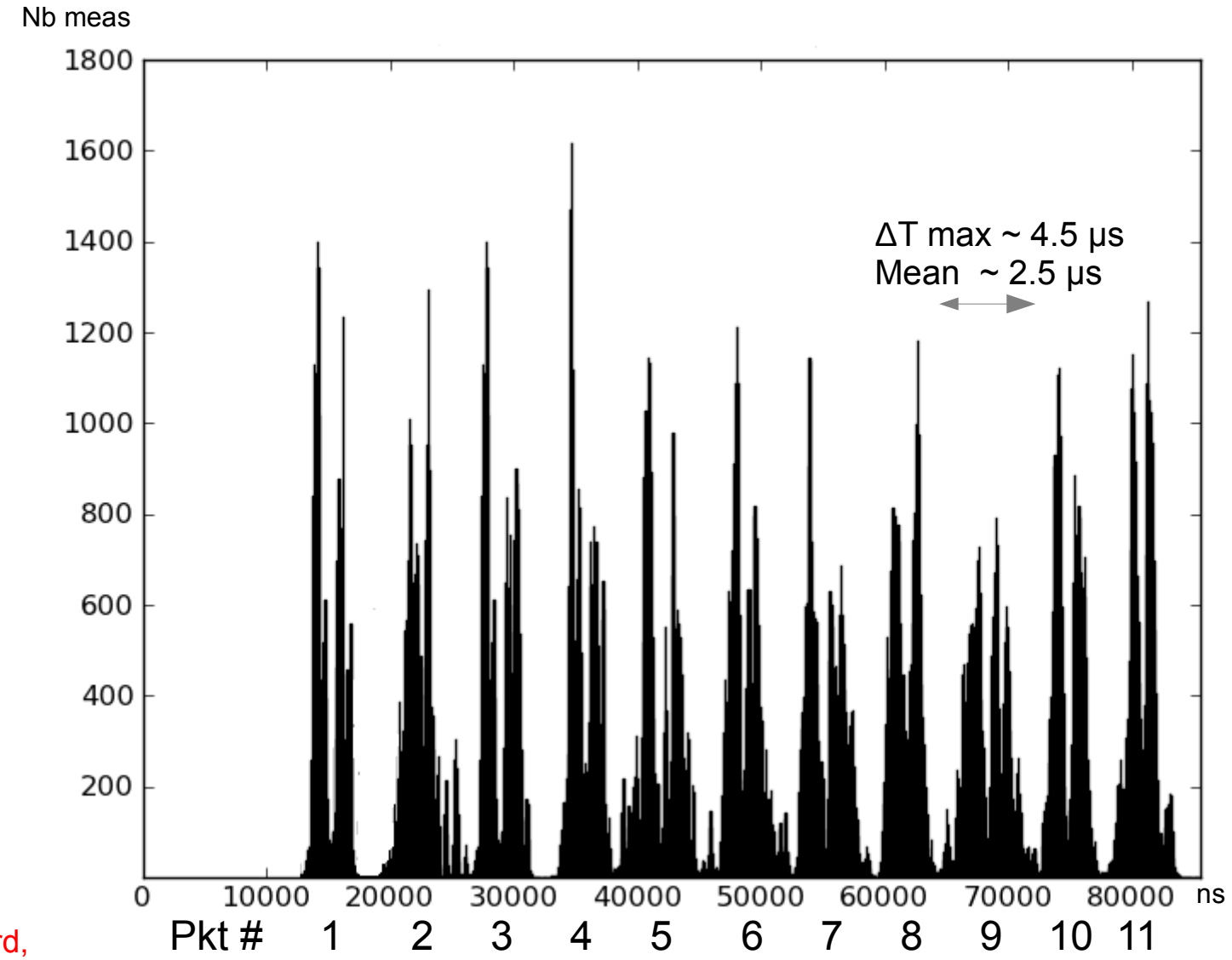


With EVOC Board,
Results slightly better with Jetway board

Stimulator : Board evaluation

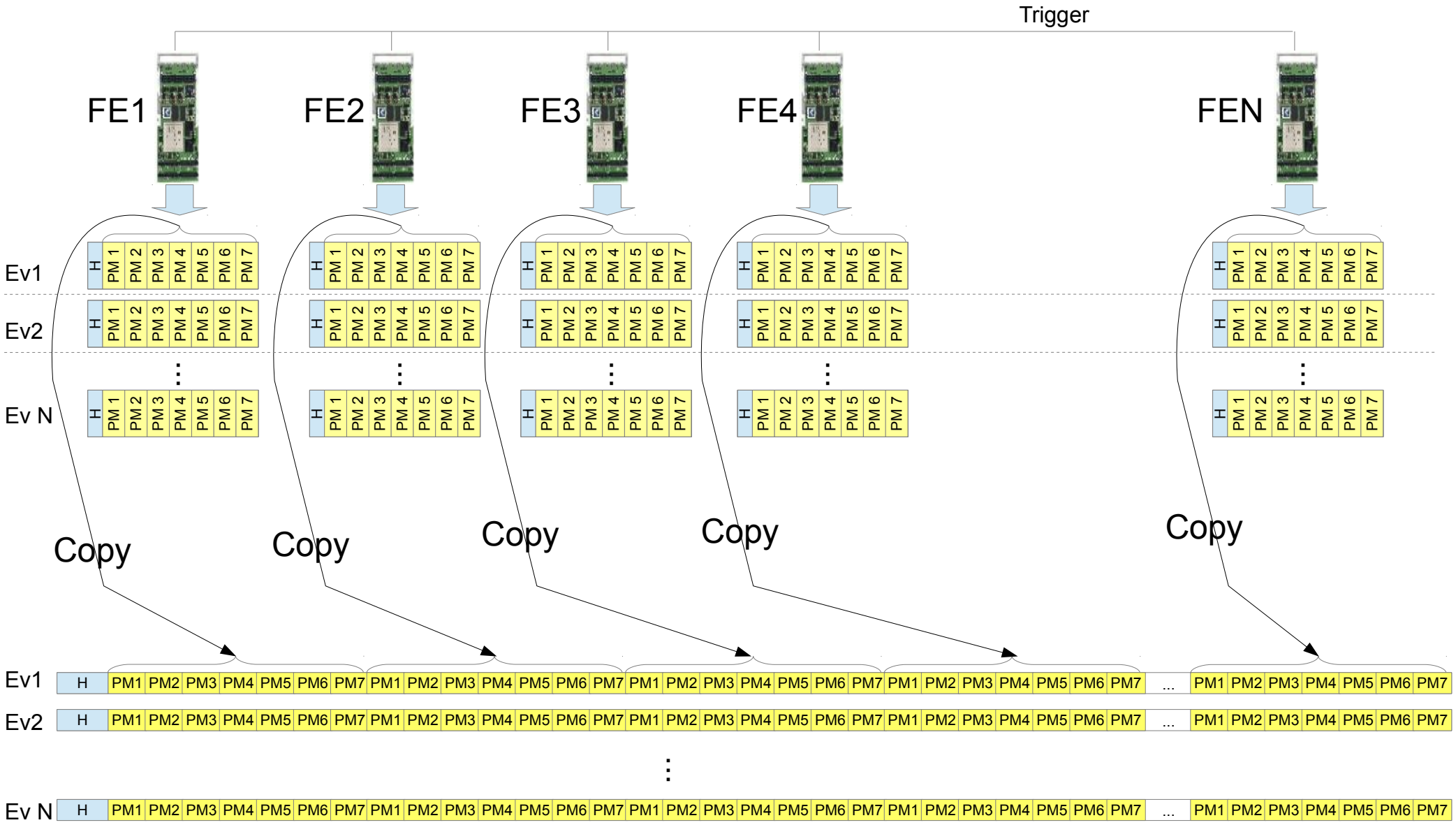
Sending of 11 consecutive packets on each port :
First and last sending ports on one SBC

10000 measurements **3 ports** 11 packets per port

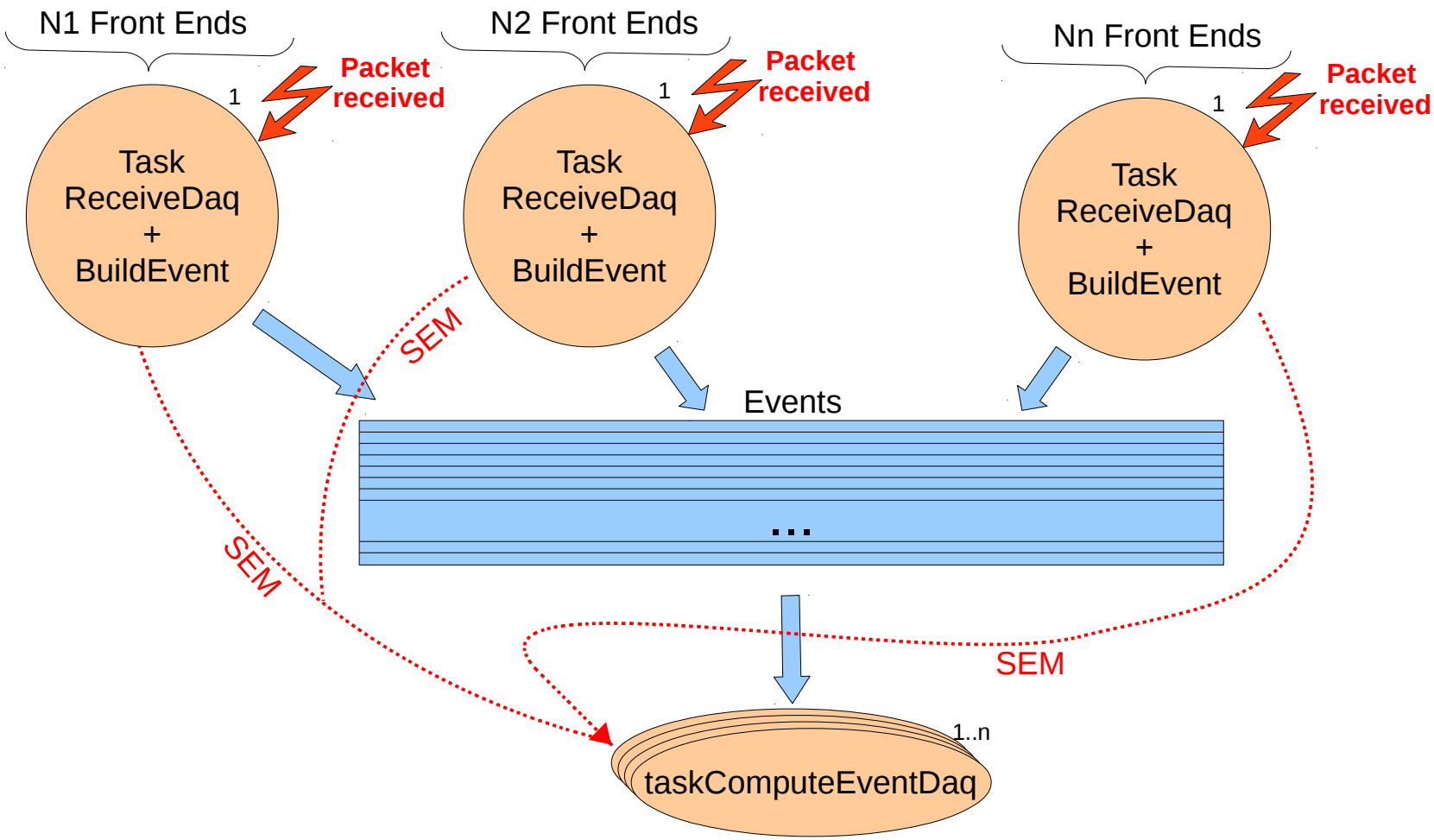


With EVOC Board,
Results slightly better with Jetway board

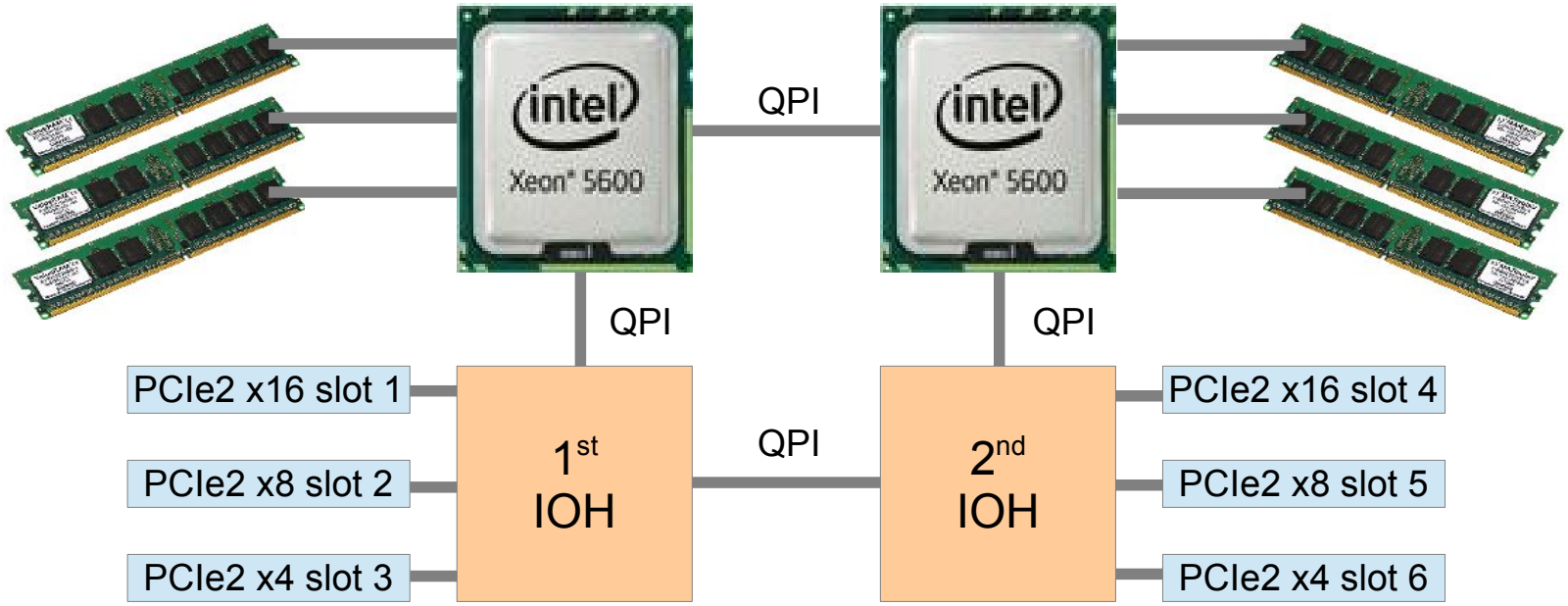
Event builder



Software overview : 2nd architecture



Non Uniform Memory Access



QPI @ 6.4 GT/s bandwidth < DDR3-1333 memory bandwidth