May, 14 2015

OSNT – A Community-owned platform for high-performance and low-cost network testing

Gianni Antichi

gianni.antichi@cl.cam.ac.uk

Let's start with the context.

We all know why we need network testers. We also know there are plenty of solutions that do the job.

Why OSNT then?

- commonly closed and proprietary systems
- limited flexibility
- well outside the reach of most universities and research laboratories

Introduction

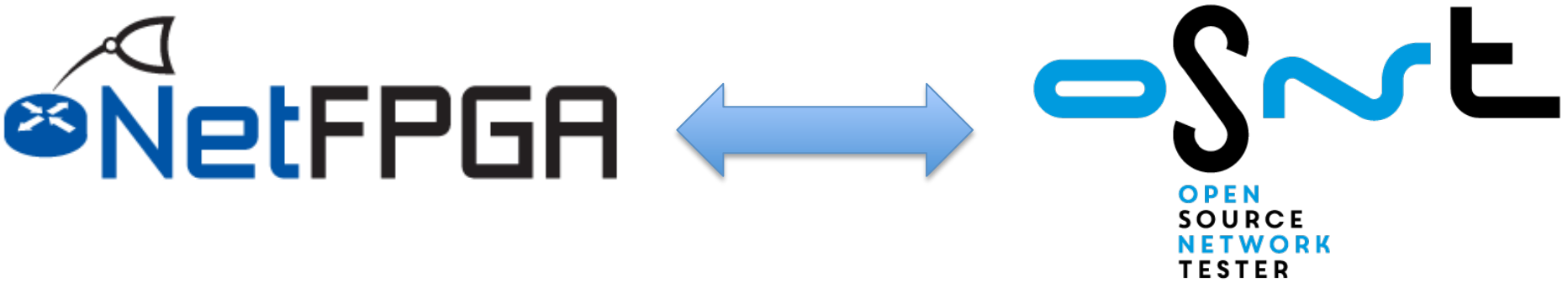
Long development cycles and high cost create a requirement for open-source network testing

- **Open-source hardware/software co-design**
- **For research community**

- flexible
- scalable
- community-based

www.osnt.org

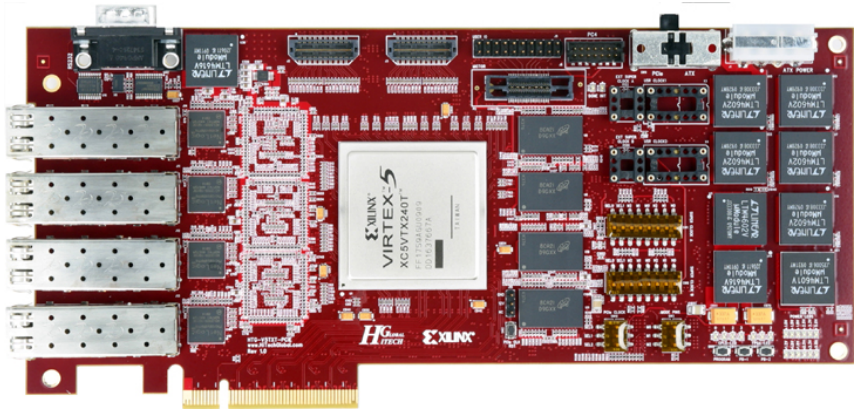
Introduction



- NetFPGA platform enabled the first prototype of OSNT.
- The open nature of NetFPGA ecosystem represents the best starting point for open HW/SW community-oriented projects.
- OSNT aims to build a community as NetFPGA did.

NetFPGA-10G

NetFPGA (Networked FPGA) is a line-rate, flexible, open networking platform for teaching and research



NetFPGA

- NeFPGA board
- tools + reference designs
- contributed projects
- community

OSNT architecture on NetFPGA-10G

OSNT flexibility provides support for a wide range of use-cases

- **OSNT-TG**
  - a single card, capable of generating packets on four 10GbE ports
  - to test a single networking system or a small network

- **OSNT-MON**
  - a single card, capable of capturing packets arriving through four 10GbE ports
  - to provide loss limited capture system with both high-resolution and high precision timestamping

OSNT architecture on NetFPGA-10G

- **Hybrid OSNT**
  - the combination of Traffic Generator and Traffic Monitor into single FPGA device and single card
  - to perform full line-rate, per-flow characterization of a network (device) under test

- **Scalable OSNT**
  - our approach for coordinating large numbers of multiple generators and monitors synchronized by a common time-base
  - still largely under work

OSNT-TG

- PCAP replay function
  - replay traces at a configurable rate
  - look at it as a sort of Hardware-Assisted “tcpreplay”
  - PCAP traces dumped straight in the SRAM (27 MB)

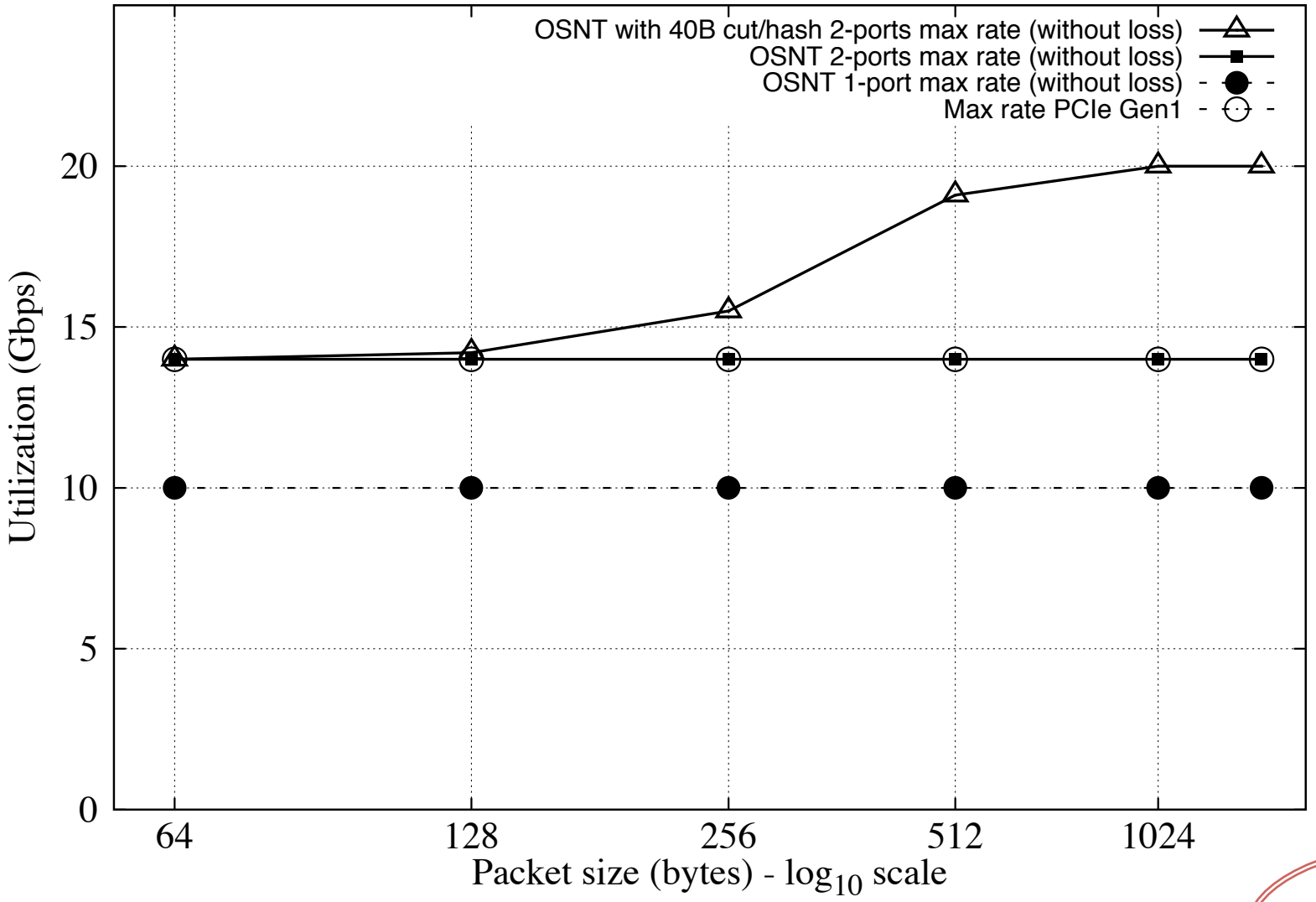
- Possibility to embed hardware TX timestamp alongside a packet count at a configurable offset
  - timestamp can be GPS corrected

- full line rate regardless packet length on 2 ports (10Gbps ports)

OSNT-MON

- Packet capture functionality
  - GPS corrected hardware timestamp in reception
- Two traffic-thinning approaches
  - hardware packet filtering (5-tuple)
  - hardware snap-length (hash for the discarded part)
- High level traffic statistics being calculated in hardware
- Libpcap patch for nanosecond granularity provided

OSNT-MON evaluation



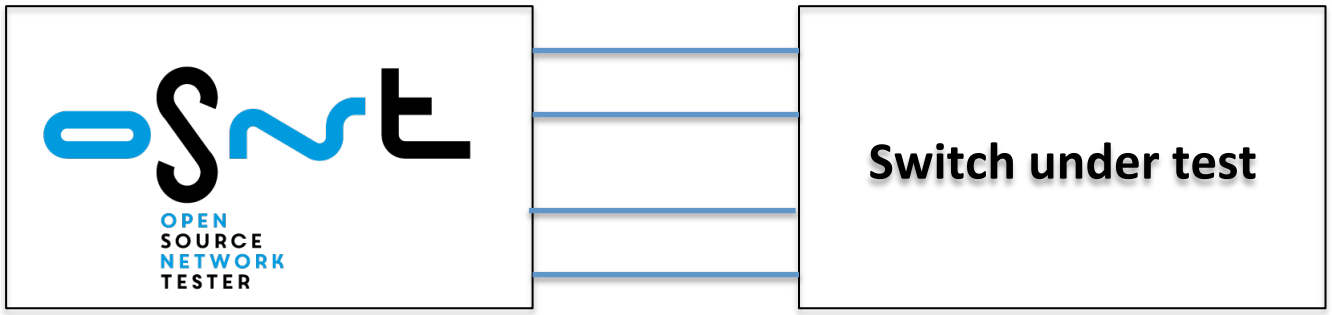
what can we do from here?

how can we effectively use OSNT?

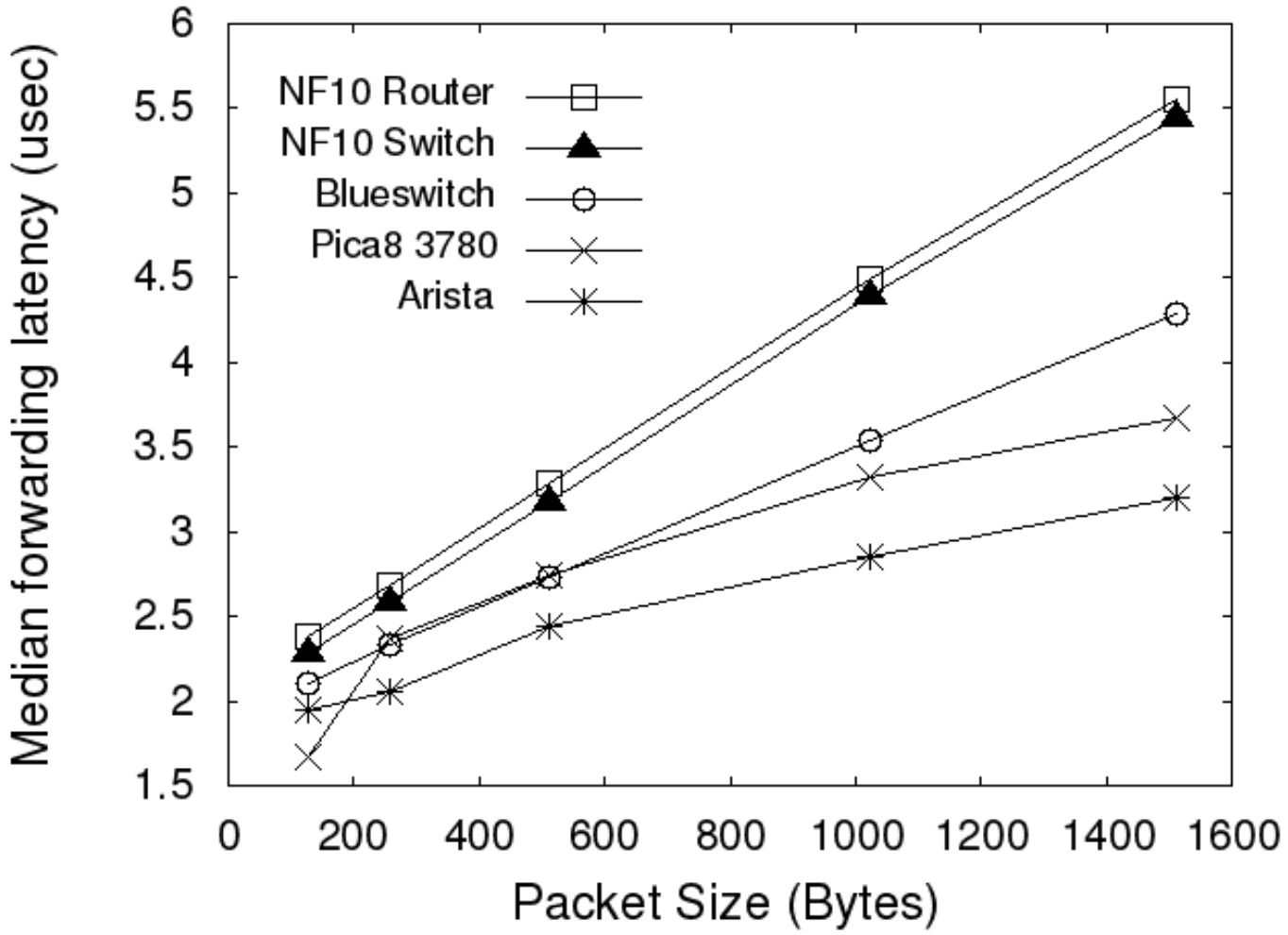
- traffic characterization (OSNT is an high precision traffic capture system)
- networking device testing (OSNT is an high performance traffic generator)
- adapt OSNT to your needs (OSNT is open, OSNT is a starting point)
- **<u>What about using OSNT for switch performance evaluation/characterization? (i.e., latency)</u>**

how is it possible to characterize a networking device latency with OSNT?

- we can embed the transmission timestamp into the packet
- OSNT can send packets at high rates and wait them back
- Compare the TX timestamp with the RX one.



woooot!!!!! I can accurately measure switching latency!



ok…this is cool, but what's next?

- participate, contribute to the open source network testing community
- extend OSNT with new features

yes, ok..but…

- **Where can we go from here?**
- **How can we fully exploit OSNT?**

the effective integration of the OpenFlow protocol in production requires a flexible and high-precision open-source measurement platform which provide a deep understanding of switch capabilities

OFLOPS-Turbo

(Open Framework for OpenFlow Switch Evaluation)

- **NetFPGA** enables **OSNT**
- **OSNT** enables **OFLOPS-Turbo**

OFLOPS-Turbo

OFLOPS

https://github.com/OFLOPS-Turbo/nf-pktgencap-lib

code available soon!!!

OFLOPS-Turbo

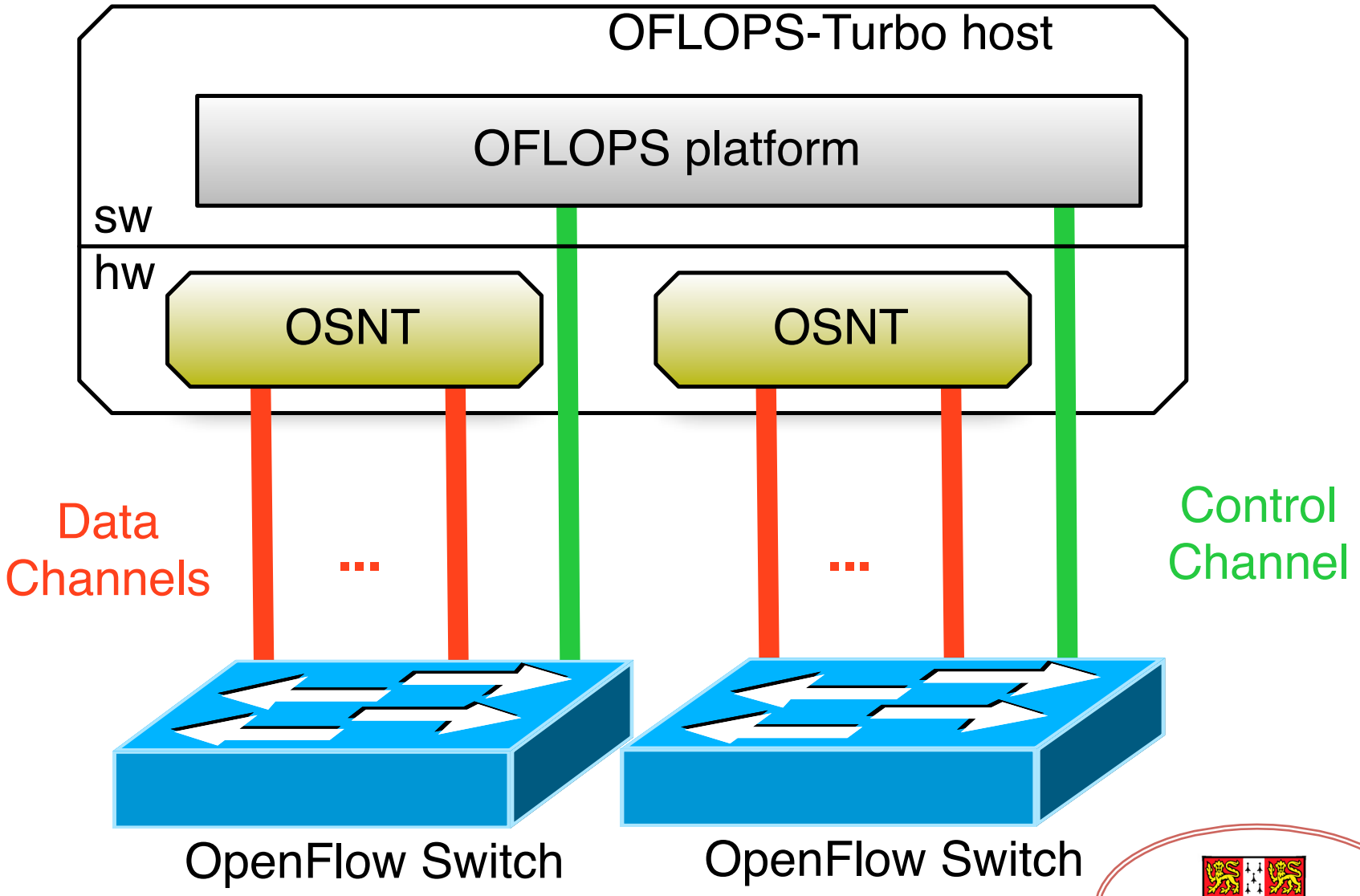
OFLOPS-Turbo

- **OFLOPS** is an holistic measurement platform which enables the development of custom OF-based experiment

- **OSNT** can be used to fulfil the data plane requirements of the experiment

- **OFLOPS-Turbo** host can be interconnected with one or more switches in arbitrary topologies and measure with high precision specific aspects of the network architecture

OFLOPS-Turbo

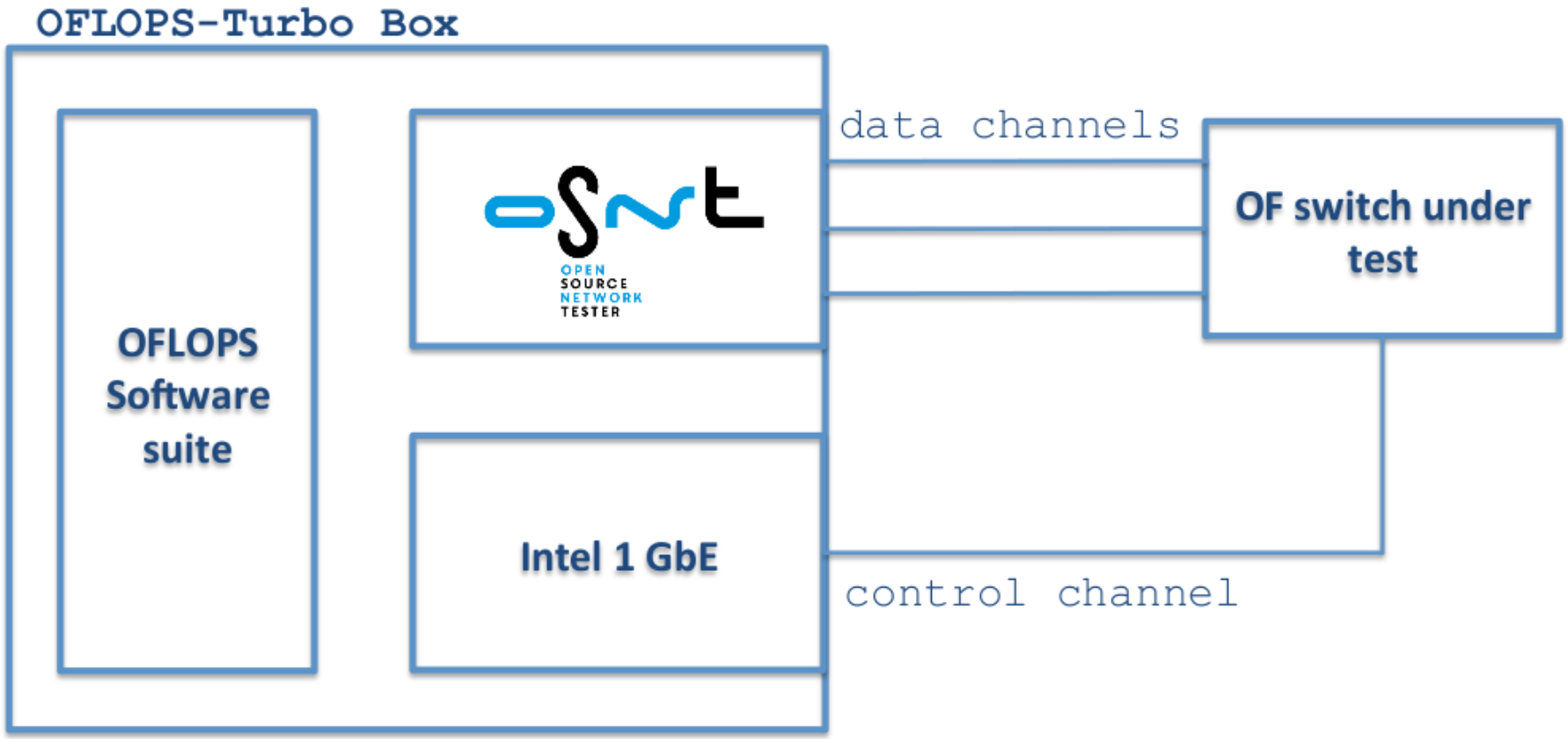


what can we do from here?

how can we effectively use OFLOPS-Turbo?

- OpenFlow flow table insertion measurements
- OpenFlow flow table modification measurements
- Create your own test in SW and test multi Gigabit switches!

Let's consider a testing scenario

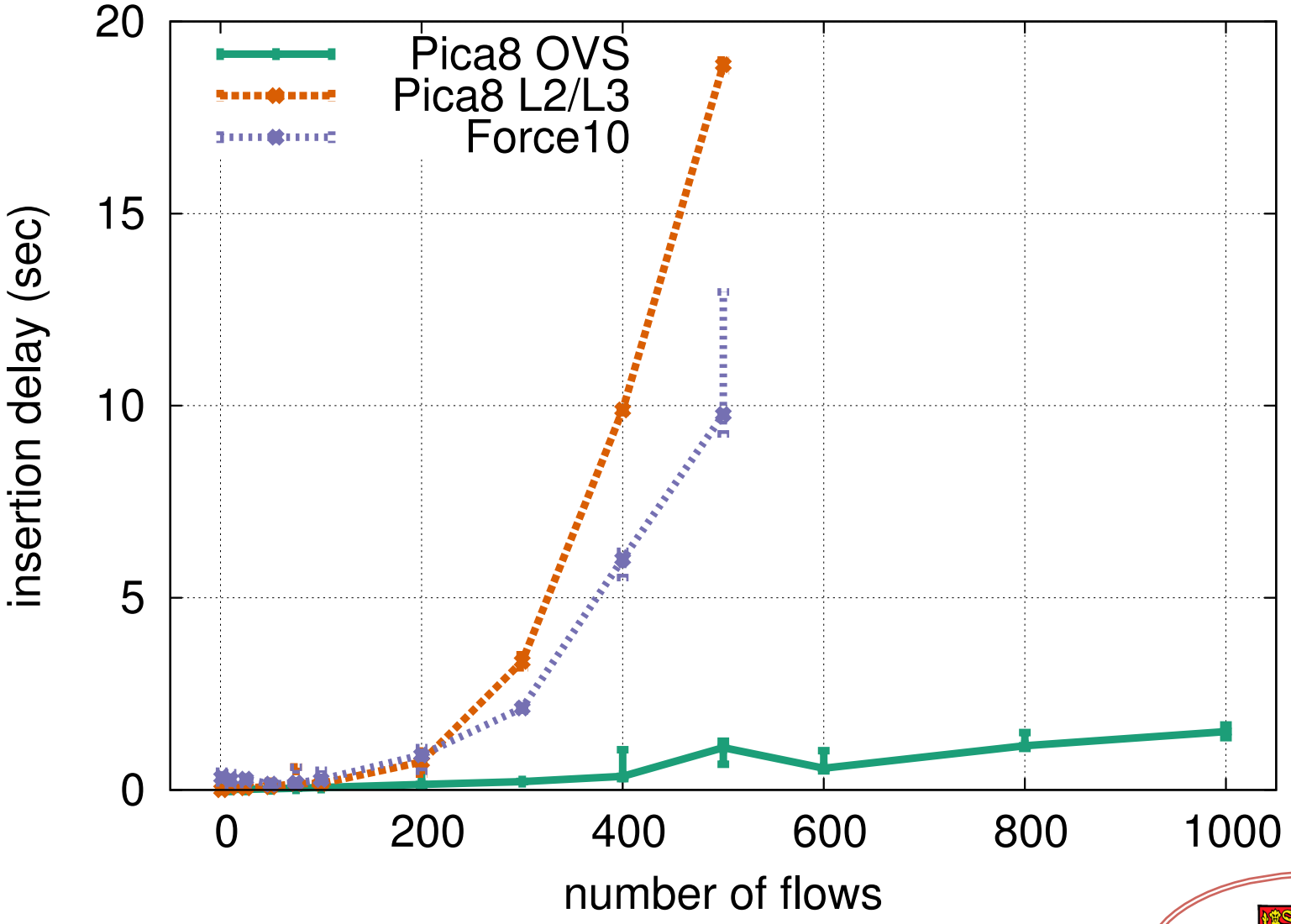


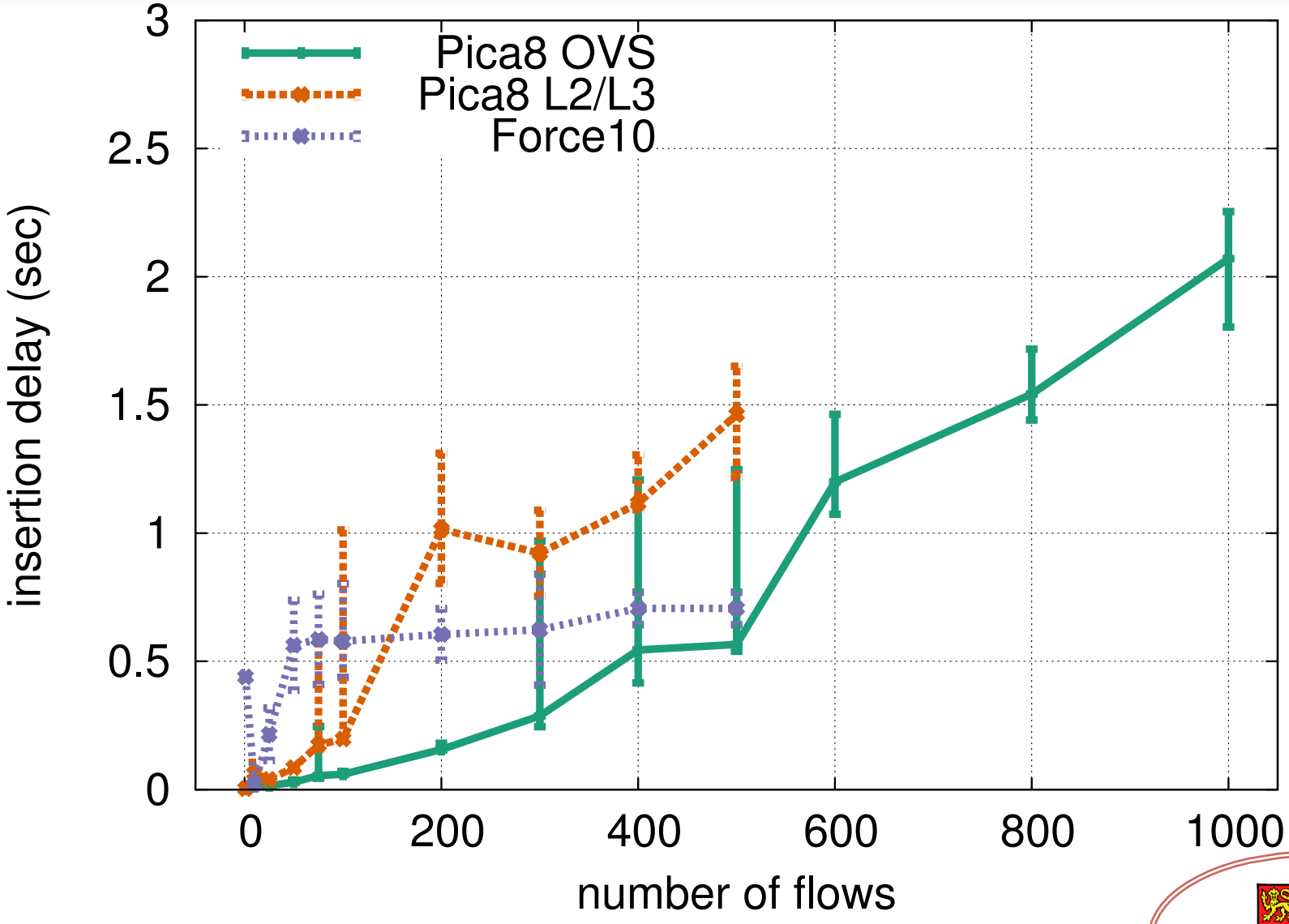
In our lab we have

01. Pica8 P3922

02. DELL Force10 S4810

Let's see what we can do with OFLOPS-Turbo!


Pica8 OVS
Pica8 L2/L3
Force10
insertion delay (sec)
number of flows
0
5
10
15
20
0
200
400
600
800
1000


Pica8 OVS
Pica8 L2/L3
Force10
insertion delay (sec)
number of flows
0
0.5
1
1.5
2
2.5
3
0
200
400
600
800
1000

this is a starting point…

OSNT/OFLOPS-Turbo are a starting point..

Questions?

Ideas?

Comments?