“Hardware and Software Design of FPGA-based PCIe Gen3 interface for APENet+ network interconnect system”

21st International Conference on Computing in High Energy and Nuclear Physics (CHEP2015)

Okinawa, 13.04.2015

Michele Martinelli

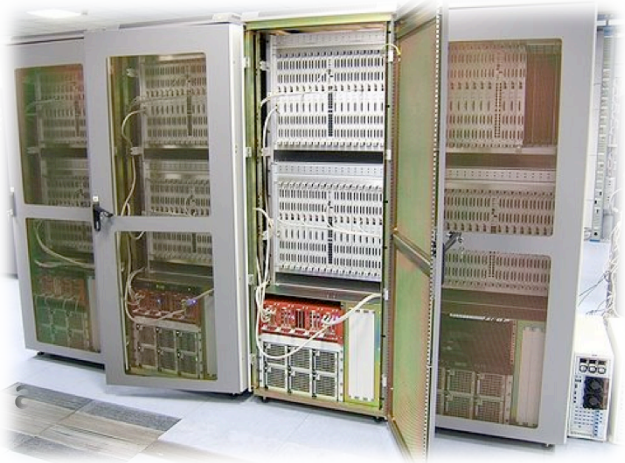
PhD Student

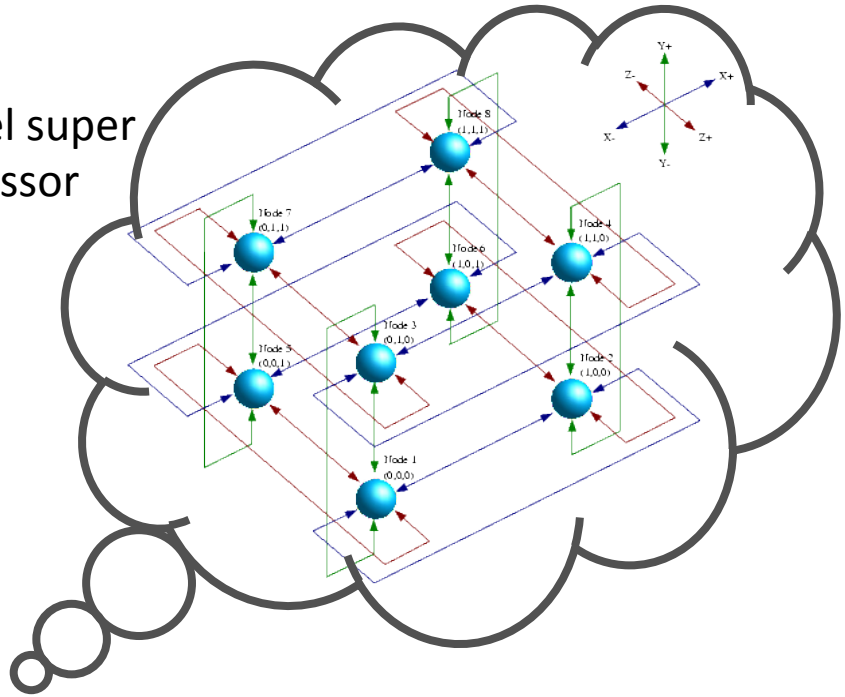
On behalf of the APE Group

INFN

INFN APE Project

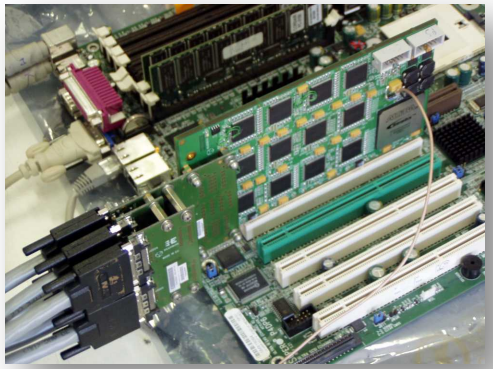
4 generations of massively parallel super computer based on custom processor



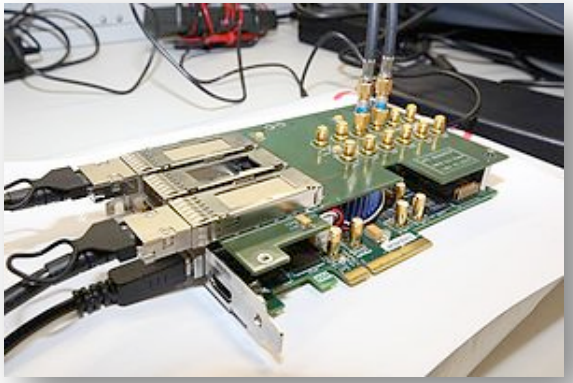


3D-clusters:
A grid with nodes arranged in a 3d torus interconnect and where each node is connected to its nearest neighbors

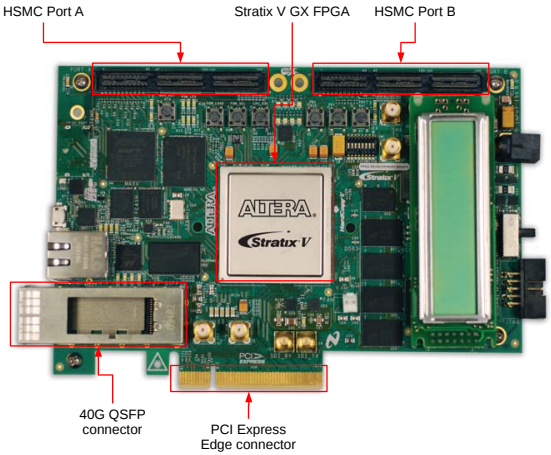
3 generations of PCI NICs for **3D-clusters** nodes interconnetion



APENet V3 (2003-2006)



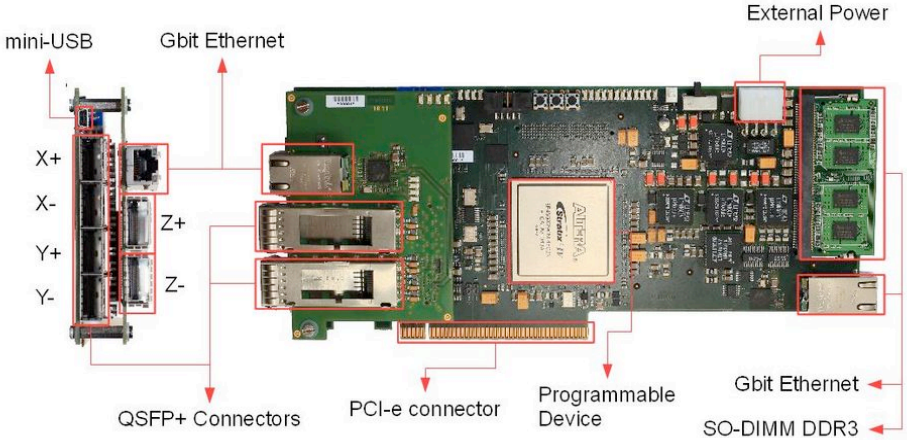
APEnet+ V4



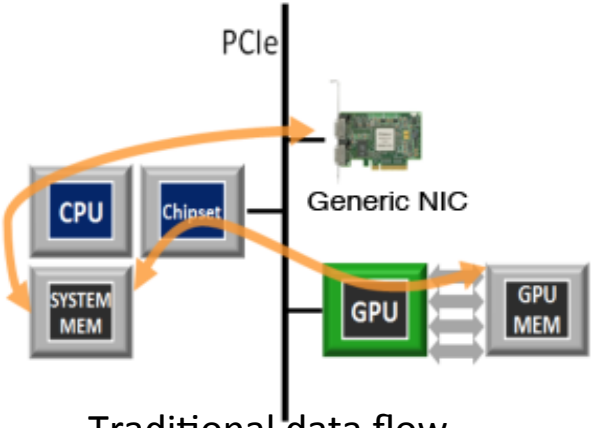
APEnet+ V5

APEnet+ V4

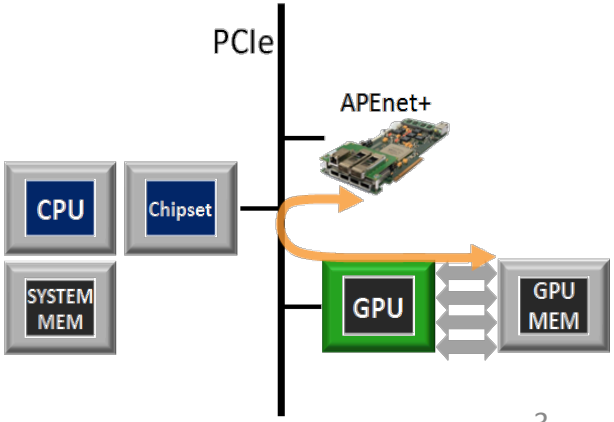
Custom board aimed for
low latency / high bandwidth interconnect



- Based on Altera Stratix IV FPGA
- PCIe Gen2 x8 (proprietary PLDA IP core)
- 6 bidirectional links directly connected to the FPGA embedded transceivers (8.5Gbps each)
  - Each link is built up by bounding 4 transceiver composing a link operating up to 34Gbps
- CPU offloading:
  - RDMA communication paradigm (real zero-copy)
    - NIOS-II microcontroller (+ TLB)
  - Nvidia GPUDirect capable
- *Used in:*
  - QUonG: HPC platform built up as a cluster of hybrid elementary computing nodes
- Related project:
  - NaNet: board more dedicated to **Real-time GPU-based** systems (see the poster!)

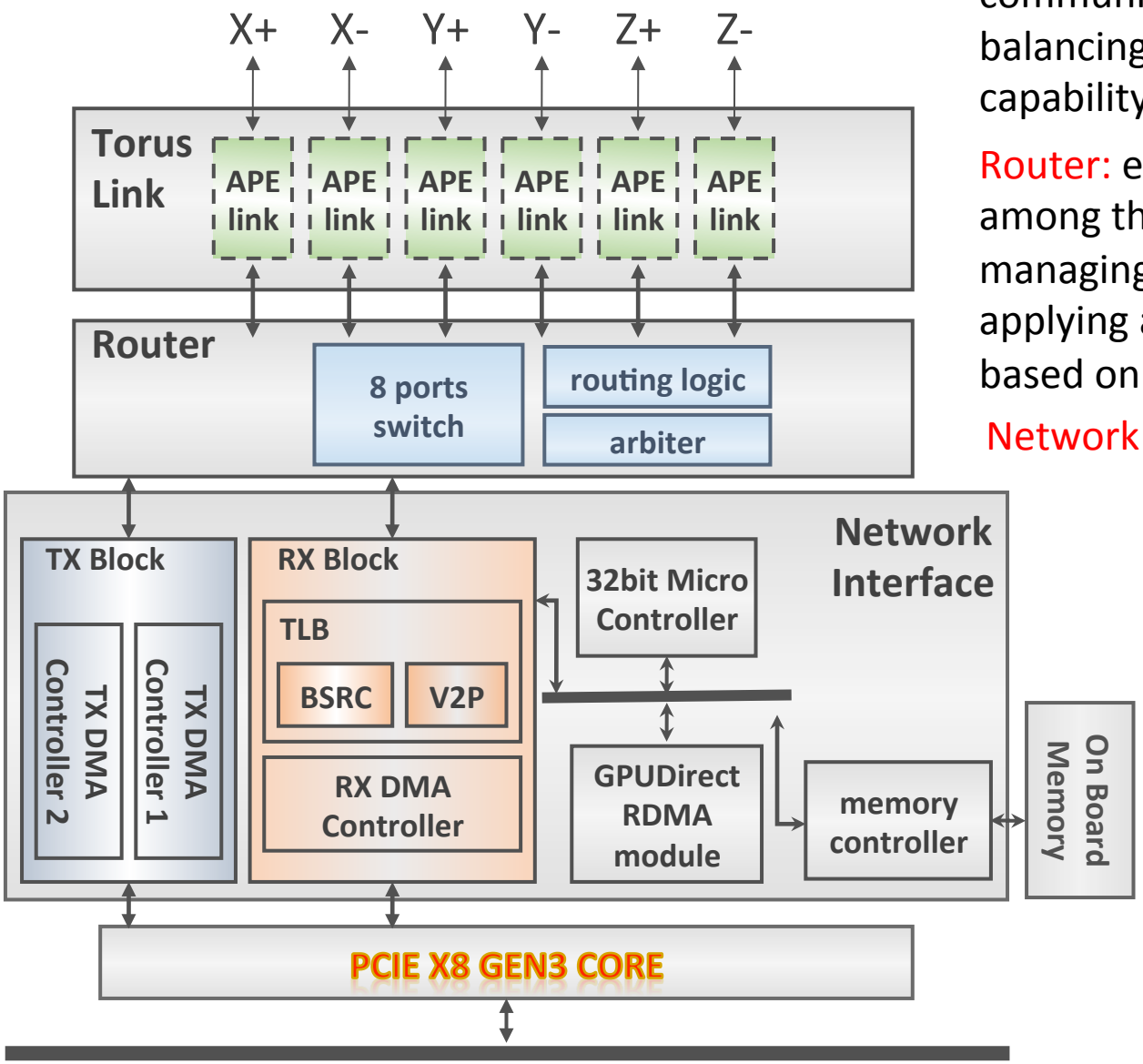


Traditional data flow



APEnet+ data flow

APEnet+ V5 HW Architecture



Torus link: node-to-node communication (error checking, DC-balancing, autonomous retransmission capability)

Router: establishes dynamic links among the 8 ports (6ext + 2int links) managing conflicts on shared resources applying a deterministic routing policy based on dimension ordering

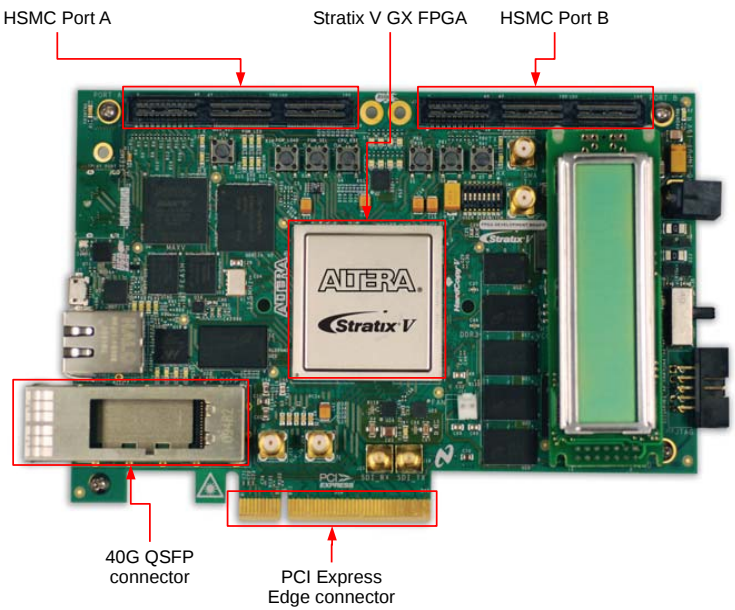
Network Interface:

- TX block: fragments data coming from PCIe port data stream into packets forwarded to the relevant destination port
- NIOS II Microcontroller/RX block: provides support for the RDMA protocol, allowing remote data transfer over the network without involvement of the CPU of the remote node

APEnet+ V5

Based on DK-DEV-5SGXEA7N dev kit:

- New 28nm Stratix V FPGA
- 40Gb QSFP+ standard interconnect fabric
- HSMC expansion ports
- PCIe connector
- 1Gbit PHY



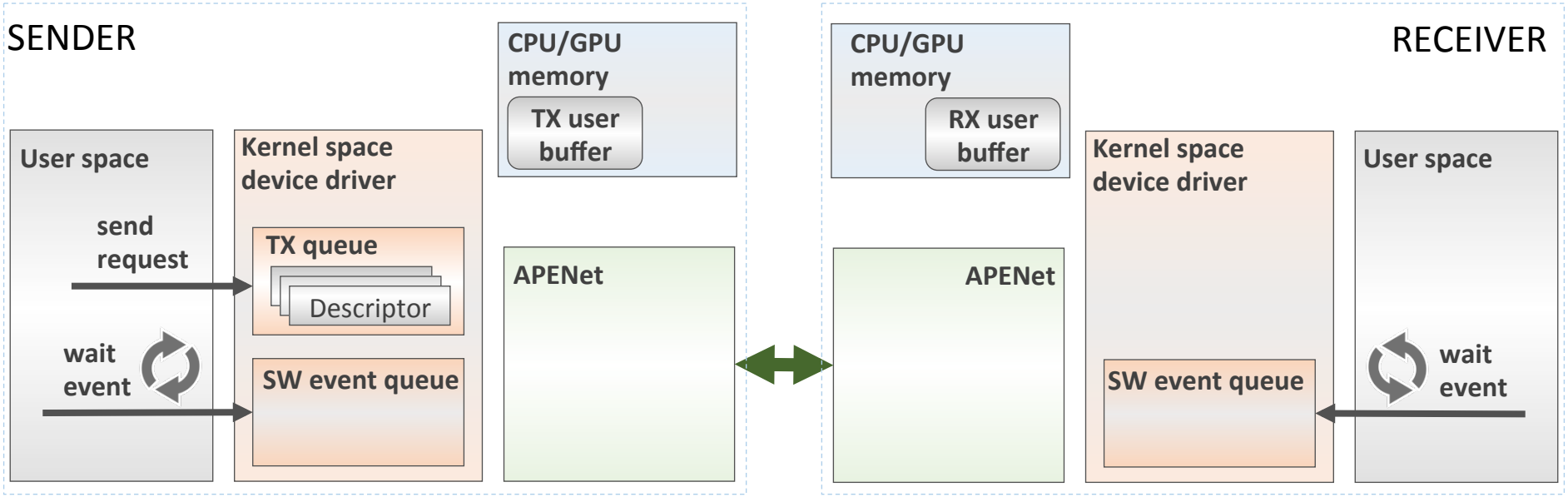
- PCIe Gen3 X8
  - Featuring 8.0 GT/s
  - Encoding scheme
    - 8b10b (20%) -> 128b130b (2%)
  - PLDA PCIe CORE IP uses axi4 interface protocol

Preliminary BER measurement

Cable	BER	Data Rate
10m Mellanox optical cable	< 2.36 E-14	11.3 Gbps
1m Mellanox copper cable	< 1.10 E-13	10.0 Gbps

- Enhanced embedded transceiver
  - up to 14.1 Gbps
    - X channel implemented using 40Gbps QSPF+ connector
      - measured at 45Gbps
    - Y/Z channels implemented on the HSMC interfaces
      - measured at 31.2Gbps/channel (to be improved)

APEnet+ V5 SW Architecture

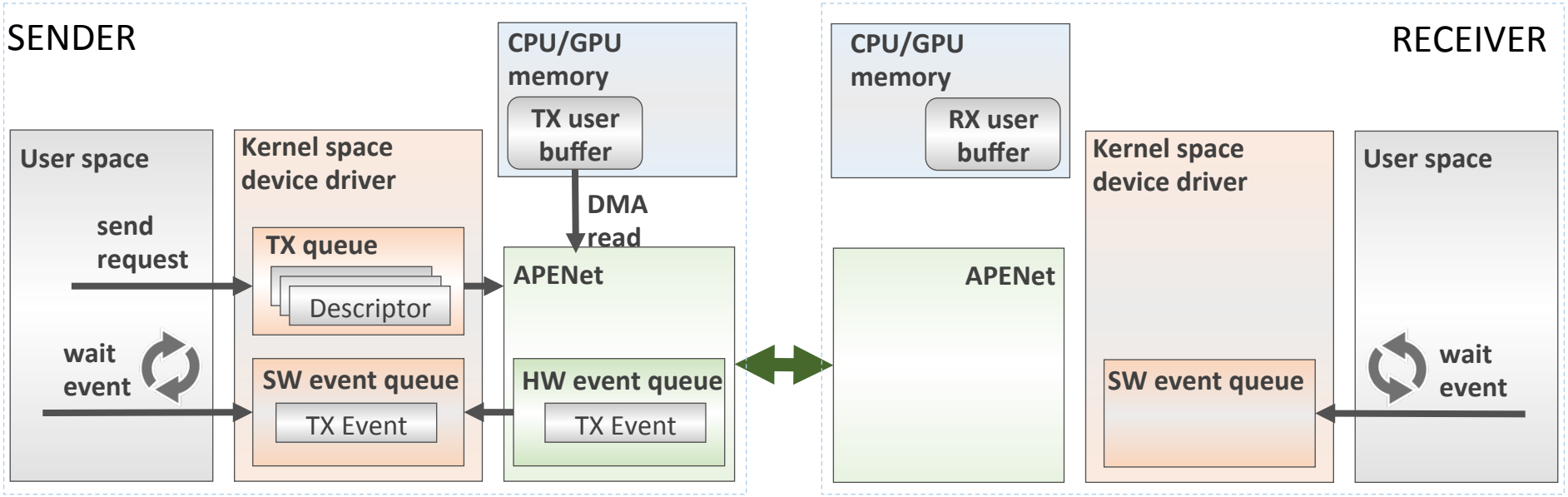


1. Buffers allocation (pin and lock memory)
   On tx side: buffer with data to transfer
   On rx side: buffer where data must be stored

2. Send request -> descriptor with TX physical address and RX virtual address

3. Wait event
   On tx side: wait for the TX_DONE event
   On rx side: wait for RX_DONE event

APEnet+ V5 SW Architecture



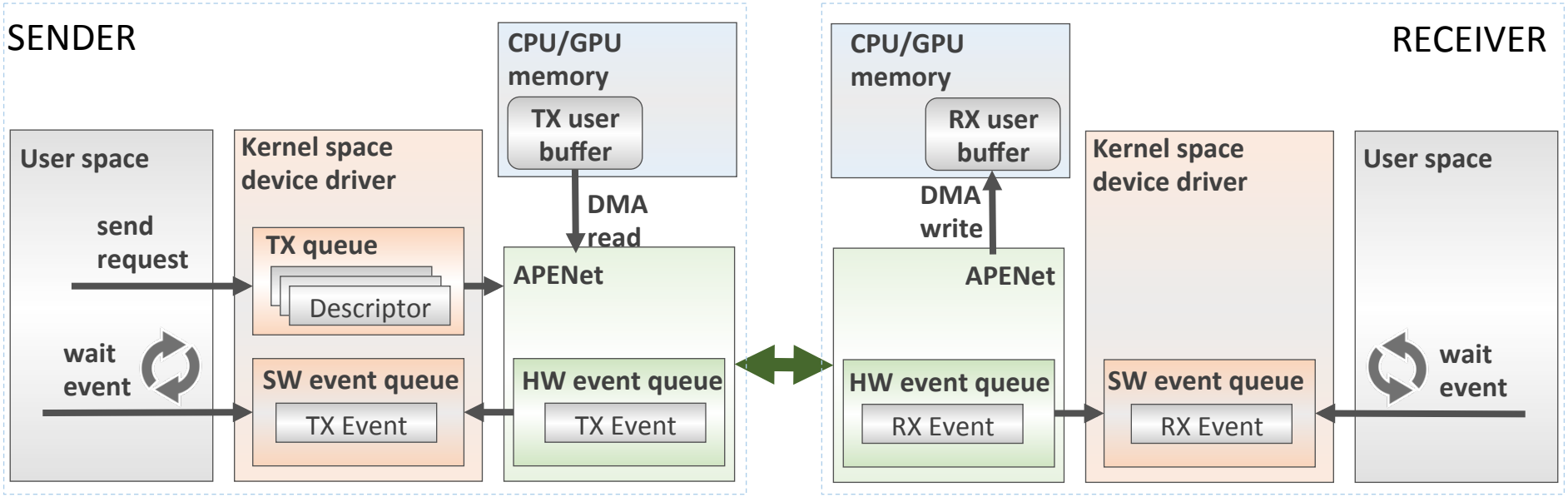
On the transmitting side

4. APENet DMA-reads data from memory (physical address)

5. Data are actually sent

6. TX_DONE event is generated and DMA-written in memory. Then it's notified to the user

APEnet+ V5 SW Architecture

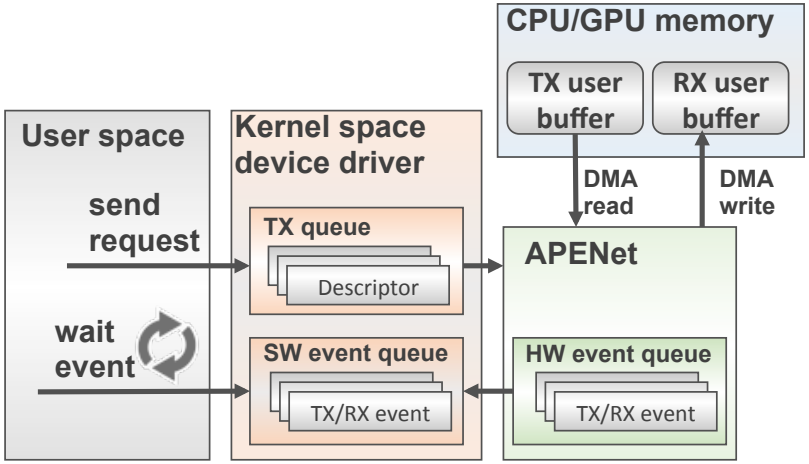


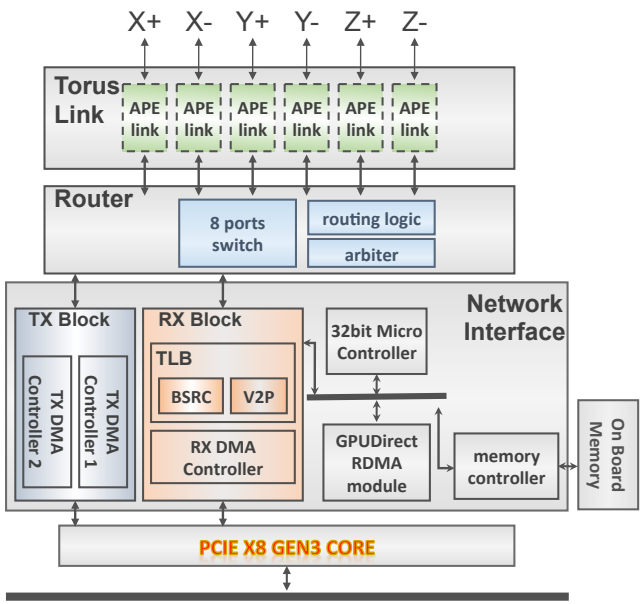
On the receiving side:

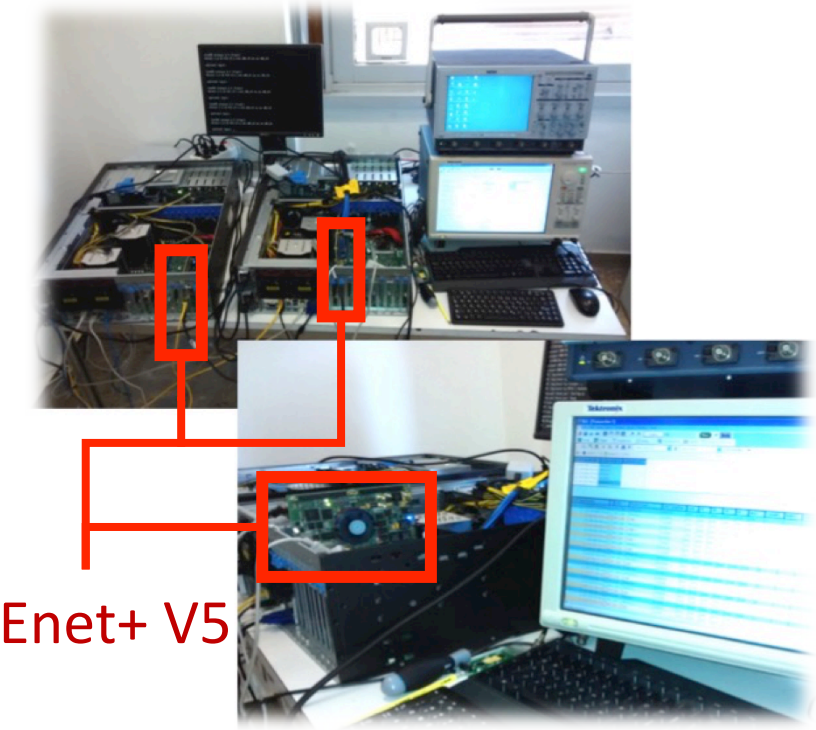
7. APENet DMA-writes data into memory (memory virtual address is translated into the correspondent physical address – CPU offloading!)

8. RX_DONE event is generated and DMA-written in memory. . Then it's notified to the user

Testbed







X9DRG-QF SuperMicro server

- Dual socket E5-2609 @ 2.40GHz
- Sandy Bridge

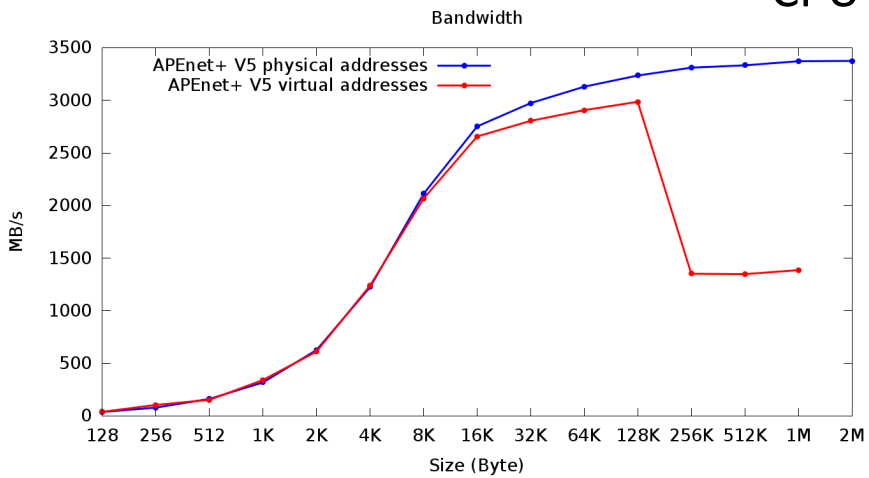
GPU Nvidia Tesla K20Xm

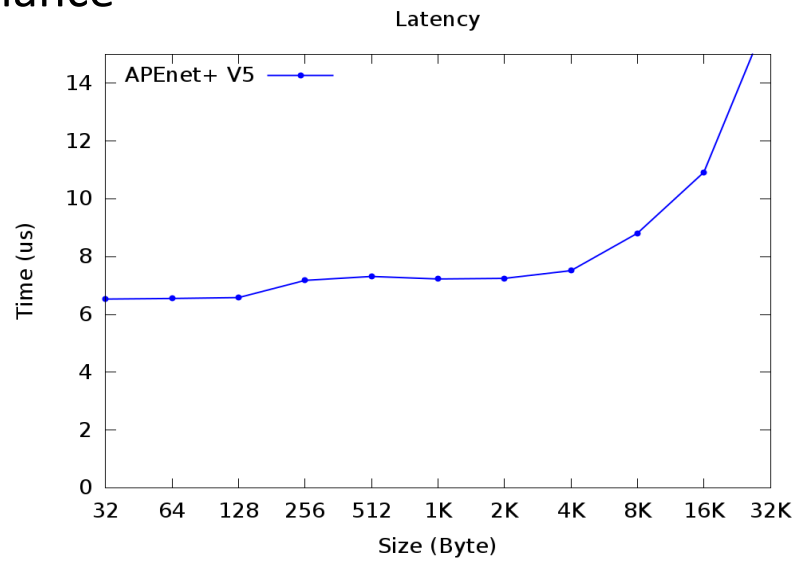
Tektronix TLA-7012 Gen3 Logic Analyzer

Latency: sending one packet of fixed size in local-loop and measuring the time required for the "sent" and "receive" events to arrive

Bandwidth: all the packets are sent in a once. Then wait for all the events

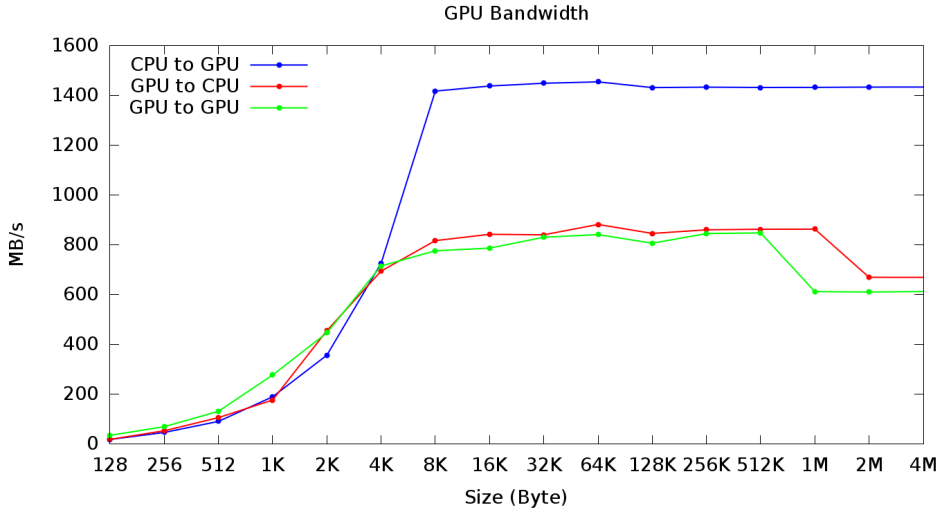
CPU performance

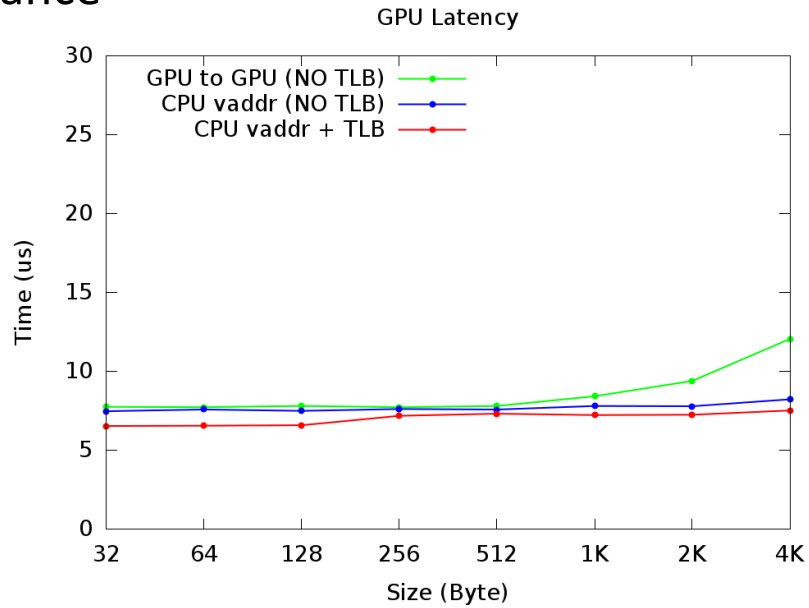




Page miss when buffer sizes exceeds the TLB capacity

GPU performance

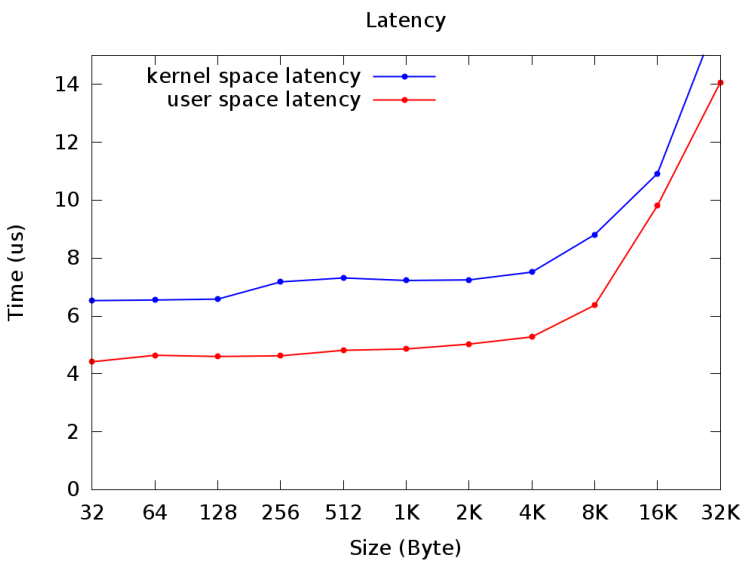




- Preliminary data (work in progress!)
- LOW GPU BW because of Sandy Bridge limitations

Work in progress…

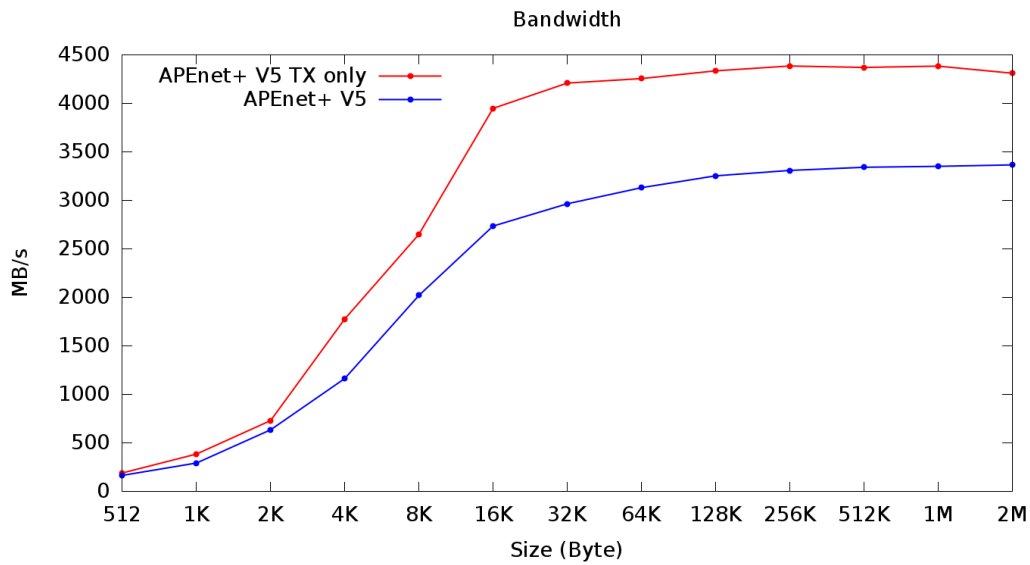
Software



User space mapping

Preliminary result on the remapping of data structure to user space

Hardware

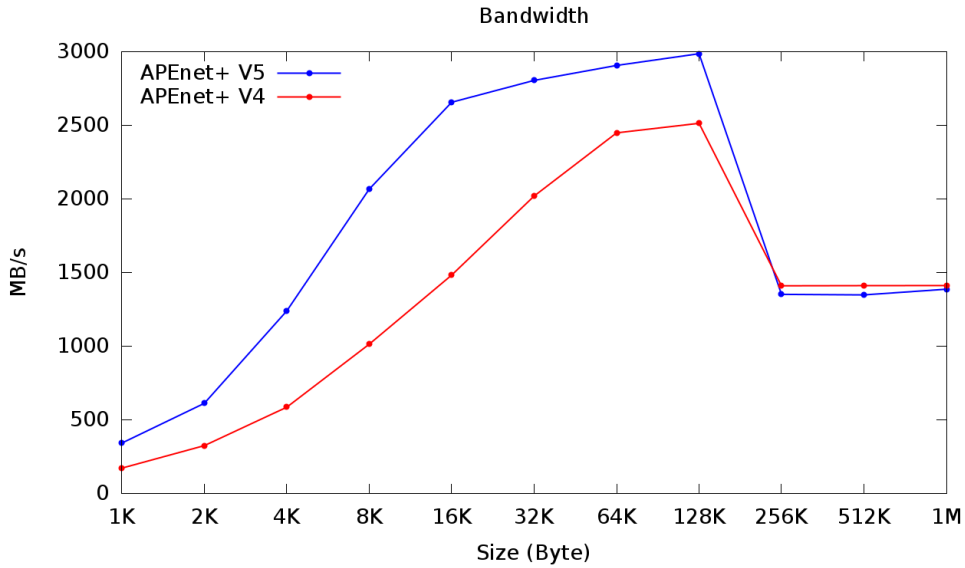


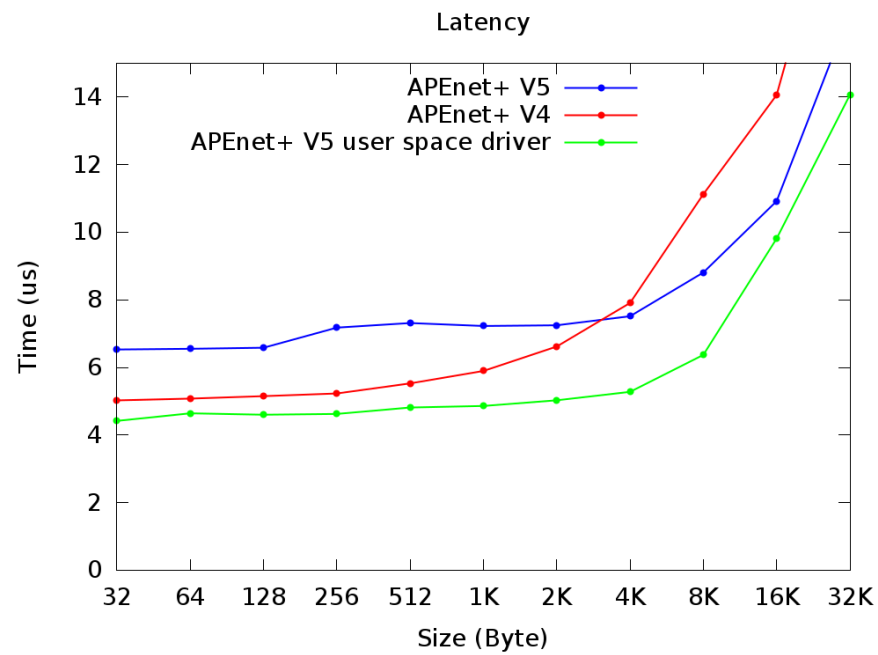
The limiting factor seems to be the RX path

Adding a second DMA engine in the RX path (as we done with the tx path) to parallelize the data transfer

Conclusions and future work

Preliminary measurement on latency and bandwidth shows improvements with respect to APEnet+ V4





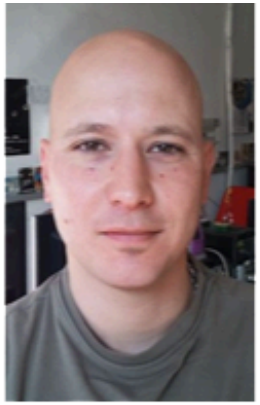
Future work:

- New testbed Ivy Bridge based to overcome the GPU BW limitation
- Evaluating the possibility of replace the NIOS-II microprocessor with an ARM/SoC

APE GROUP

Roberto Ammendola

Andrea Biagioni



Ottorino Frezza

Francesca Lo Cicero

Alessandro Lonardo

Michele Martinelli

Pier Stanislao Paolucci

Elena Pastorelli

Davide Rossetti

Francesco Simula

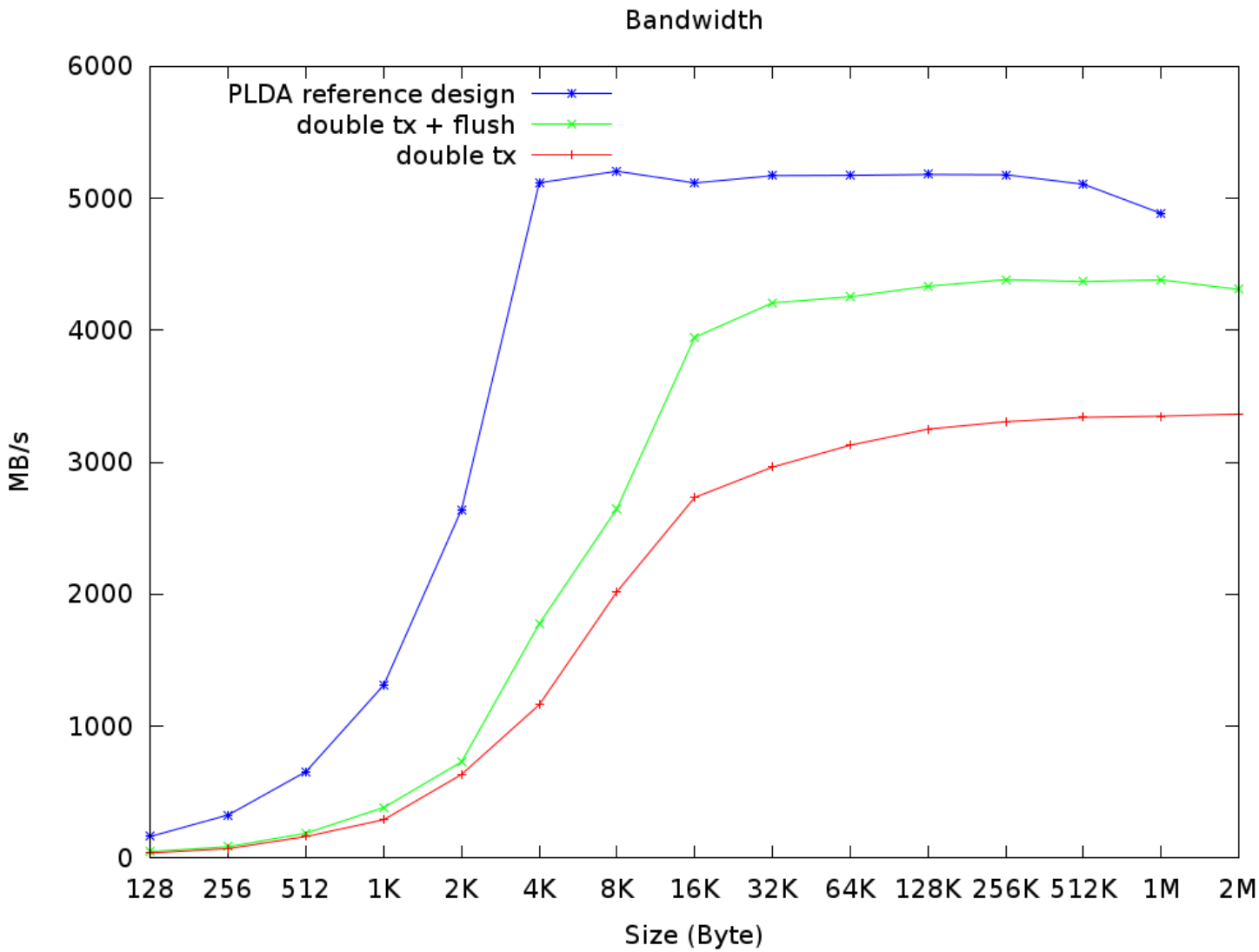
Laura Tosoratto

Piero Vicini

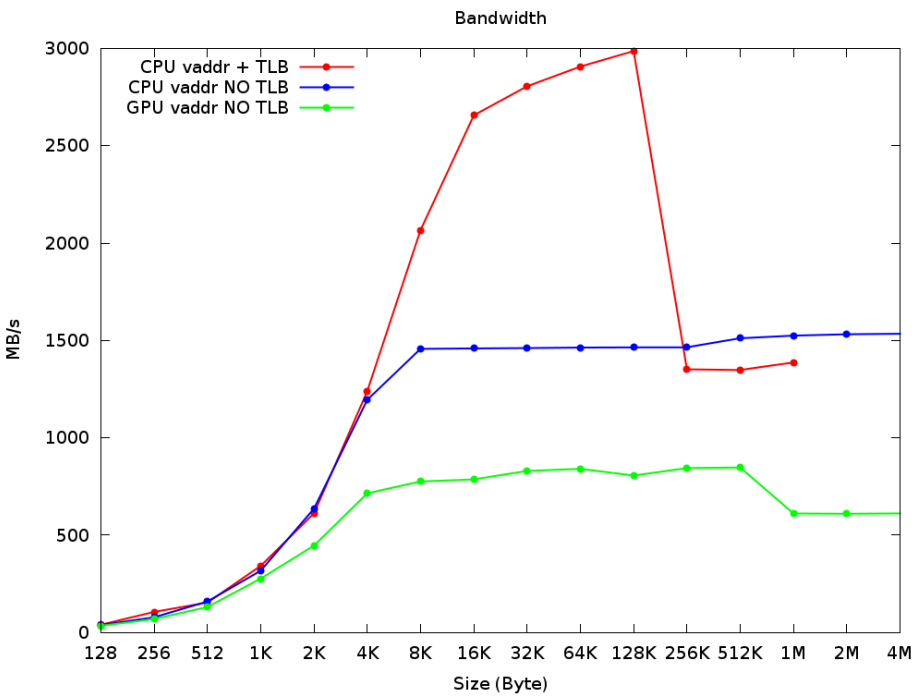
This work was partially supported by EU Framework Programme 7 EURETILE project, grant number 247846; Roberto Ammendola and Michele Martinelli were supported by MIUR (Italy) through INFN SUMA project.

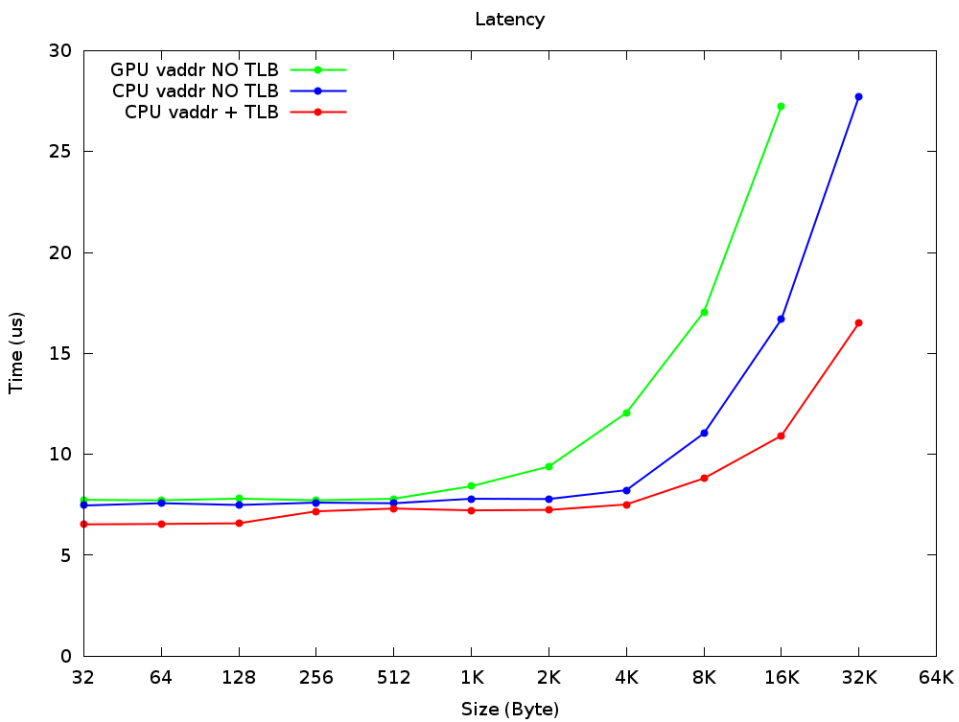
Backup slides

PLDA ref design comparison

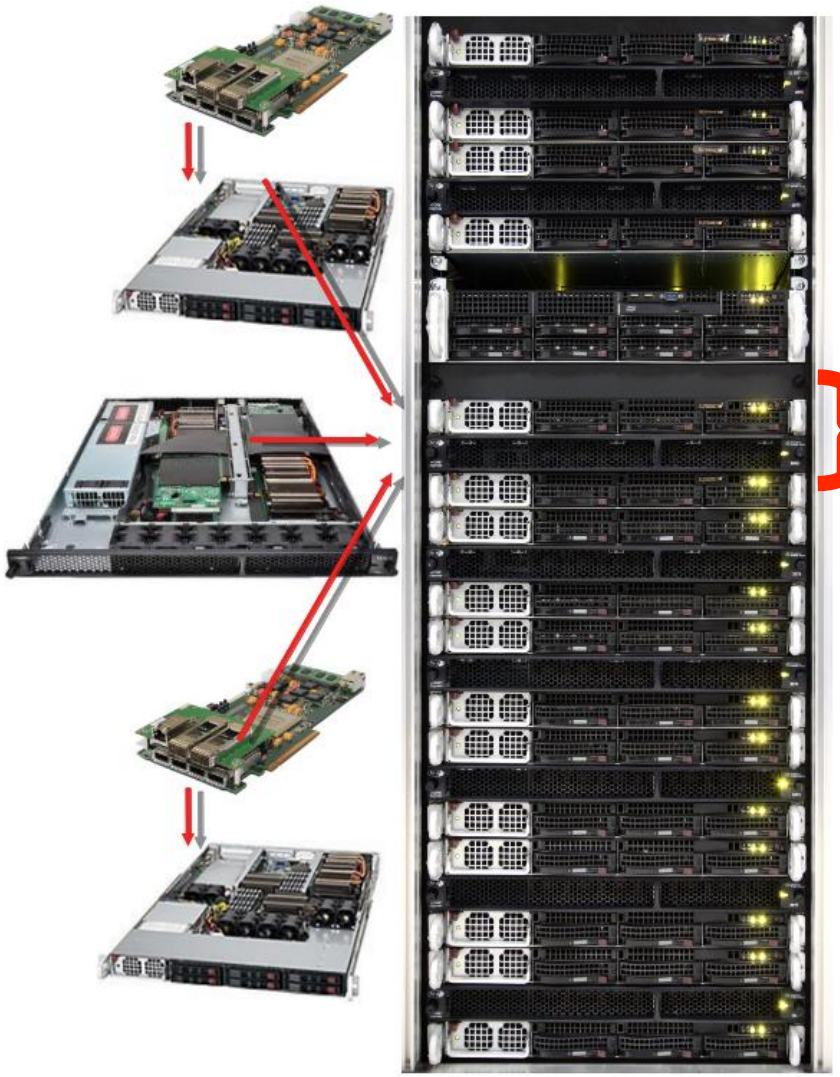


Performance- GPU results
To be improved…





QUonG: EURETILE HPC platform



16 nodes connected by APEnet+ (4x4x1)

QuonG Hybrid Computing Node:

- ✓ Intel Xeon E5620 double processor
- ✓ 48 GB System Memory
- ✓ 2 M2075 NVIDIA Fermi GPU
- ✓ 1 APEnet+ board
- ✓ 40 Gbps InfiniBand Host Ctrl Adapter

QUonG Elementary Mechanical Unit:

3U Sandwich: (2 Computing Node)

2U Intel dual Xeon servers
1U 4 NVIDIA Tesla M2075 GPU
2 Vertex on the APEnet+ 3D network

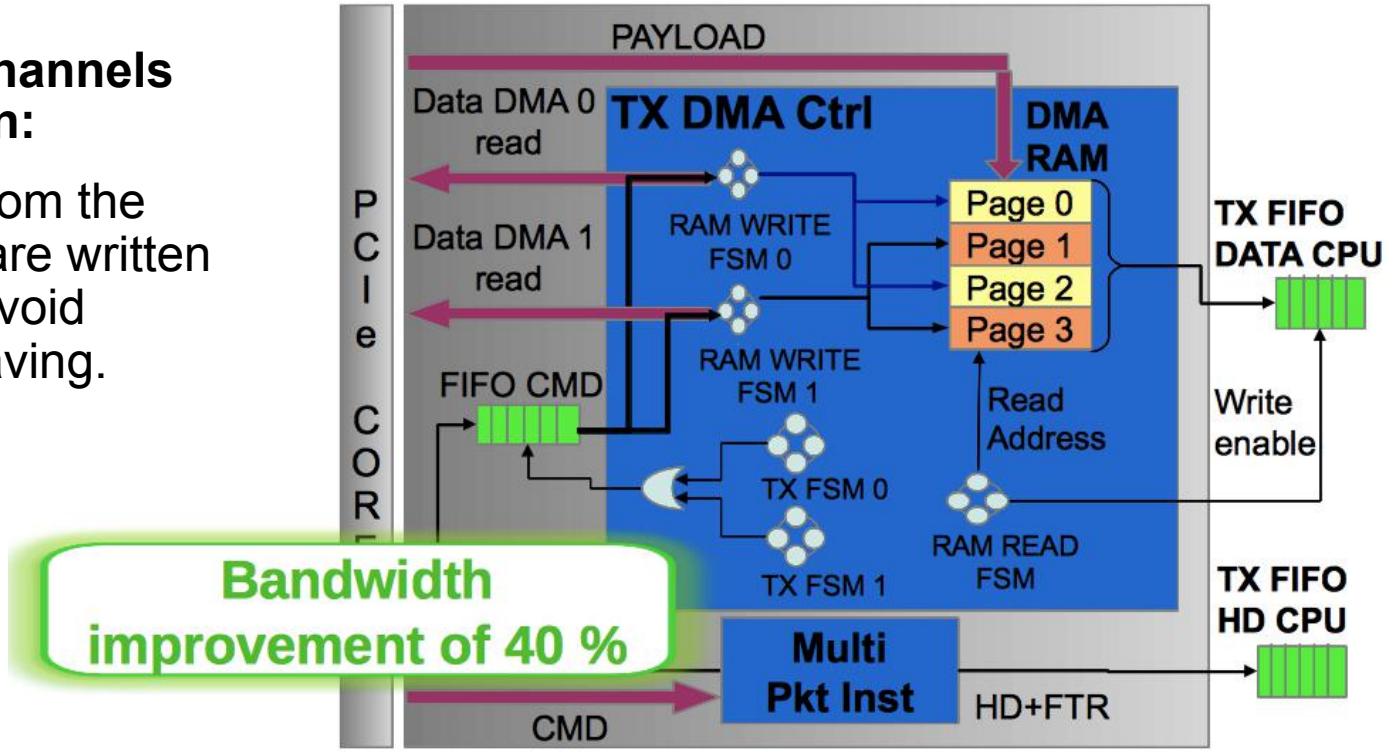
Software Environment

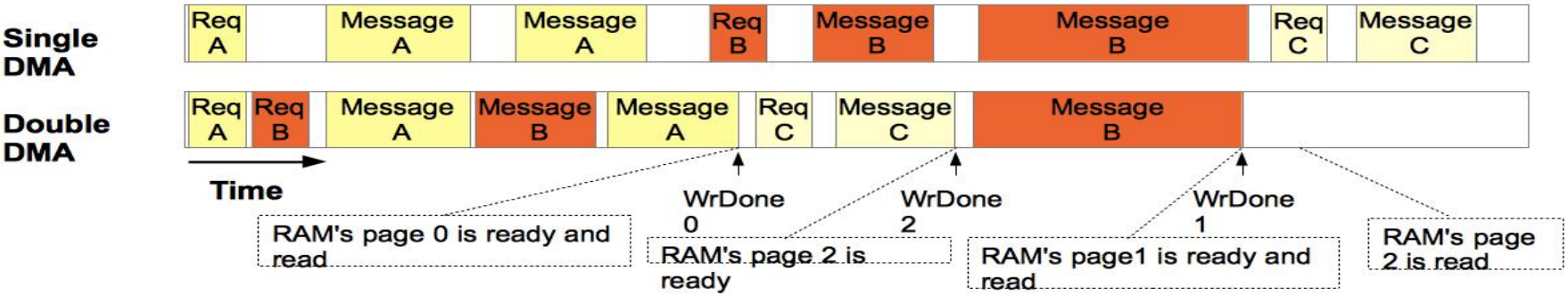
- ✓ CentOS 6.4
- ✓ NVIDIA CUDA 4.2 compilation tool
- ✓ OpenMPI and MVAPICH2 MPI available

Design improvement : Tx double DMA (II)

Double DMA channels implementation:

data received from the PCIe interface are written into a RAM to avoid payload interleaving.



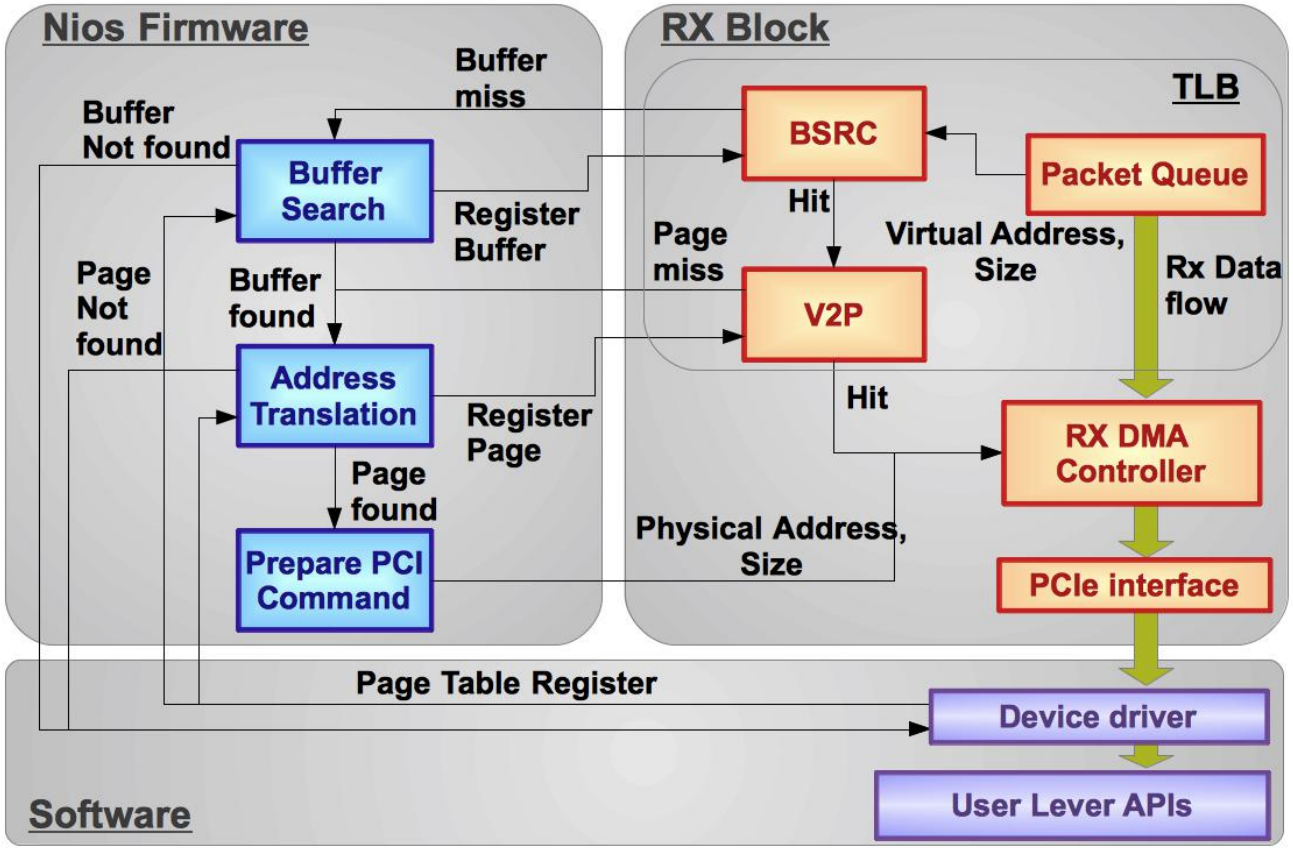


Design improvement : Rx speed up- memory management (II)

Second solution: TLB implementation

TLB (Translation Lookaside Buffer) :

- ✓associative cache;
- ✓limited amount of entries to perform memory management tasks;
- ✓reducing processing time in case of a cached translation (Hit);
- ✓forwarding the operation to the Nios in case of a 'miss';



31 clock cycles (@250 MHz) to complete buffer look-up and address translation in case of hit!