APENet: LQCD clusters a la APE

Concept, Development and Use

Roberto Ammendola

Istituto Nazionale di Fisica Nucleare, Sezione Roma Tor Vergata
Centro Ricerce "E. Fermi"

SM&FT 2006, Bari – 21th September 2006

Motivation

The APE group has traditionally focused on the design and the development of custom silicon, electronics and software optimized for Lattice Quantum ChromoDynamics.

The APENet project was started to study the mixing of existing off-the-shelf computing technology (CPUs, motherboards and memories for PC clusters) with a custom interconnect architecture, derived from previous experience of the APE group.

The focus is on building optimized, super-computer level platforms suited for numerical applications that are both CPU and memory intensive.

Requested Features

The idea was to build a switch-less network characterized by:

- High bandwidth
- Low latency
- Natural fit with LQCD and numerical grid-based algorithm.
- Not necessarily first neighbour communication.
- Good performance scaling as a function of the number of processors.
- Very good cost scaling even for large number of processors.

Development stages

APENet history:

- Sept 2001 – June 2002: Release of the first HW prototype
- Sept 2002 – Feb 2003: 2nd HW version
- March – Sep 2003: 3rd HW version development and production
- Dec 2003: Electrical validation
- Sept 2004: 16 nodes APENet prototype cluster
- March – Nov 2005: 128 nodes APENet cluster
- June 2006: Production starts

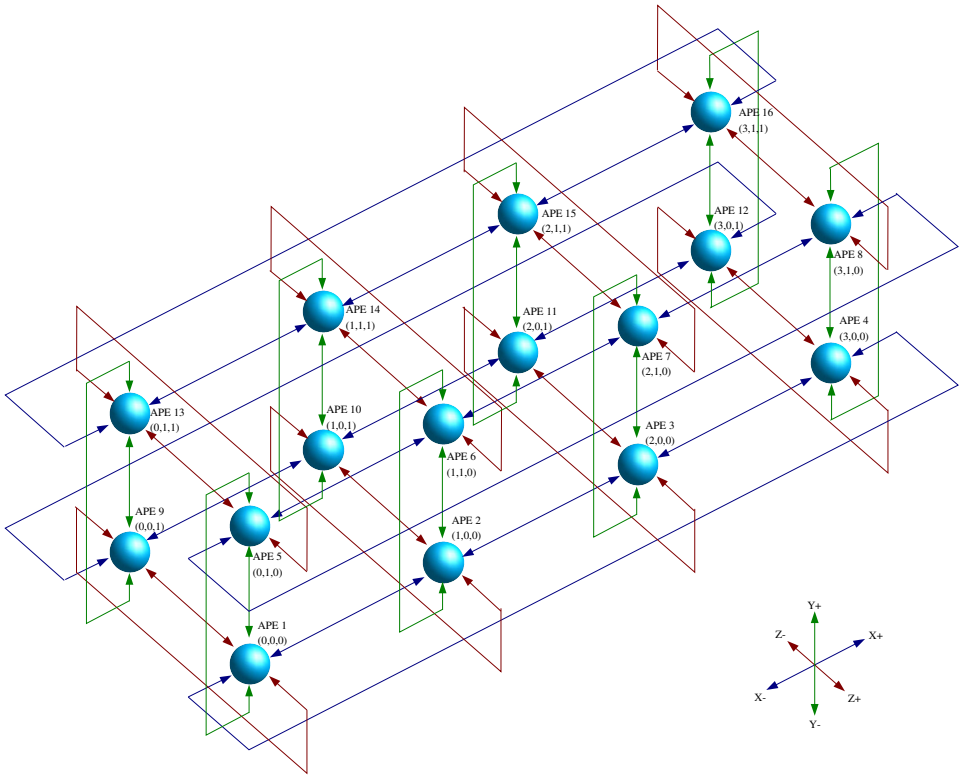
APENet Main Features

APENet is a 3D network of point-to-point links with toroidal topology.

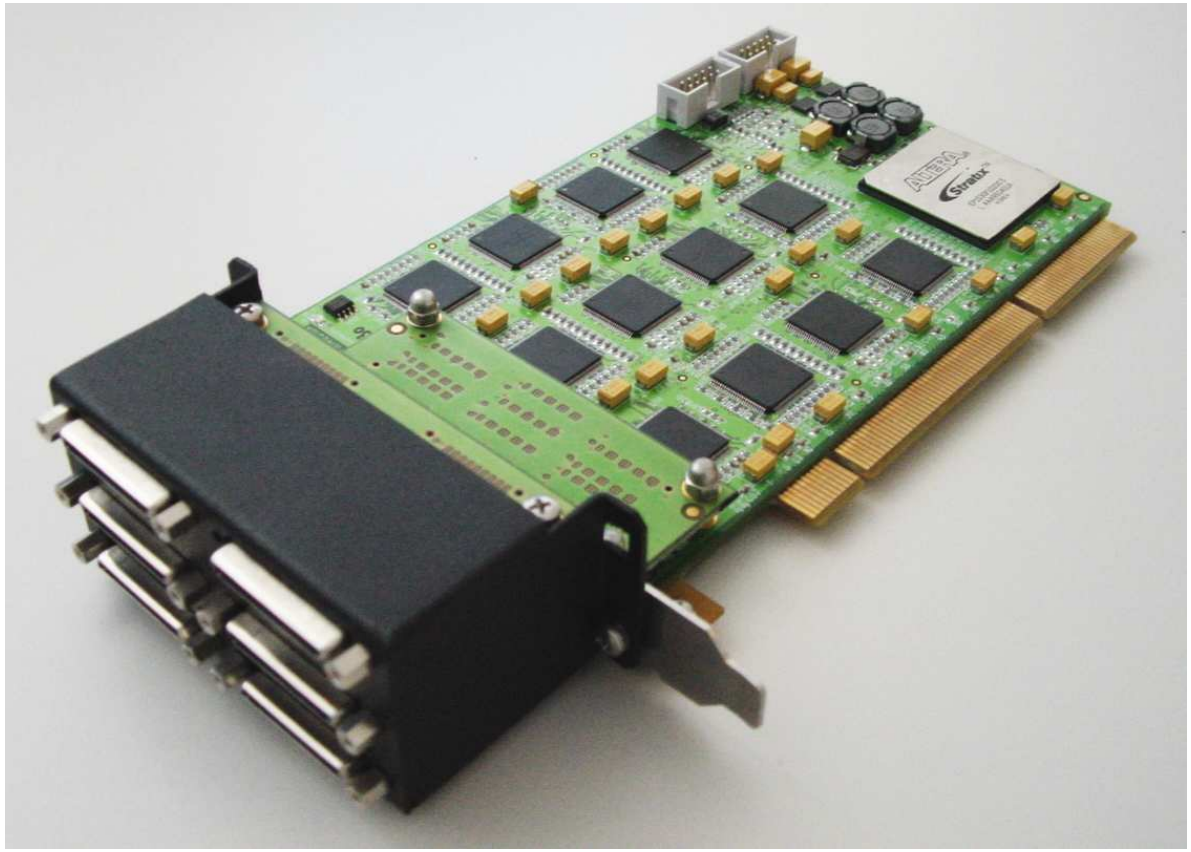
- Each computing node has 6 bi-directional full-duplex communication channels
- Computing nodes are arranged in a 3D cubic mesh
- Data is transmitted in packets which are routed to the destination node
- Lightweight low level protocol
- Wormhole routing
- Dimension ordered routing algorithm
- 2 Virtual Channels per receiving channel to prevent deadlocks

How Does it Look Like

Here it is shown a $4 \times 2 \times 2$ example in a logic description and a real implementation:



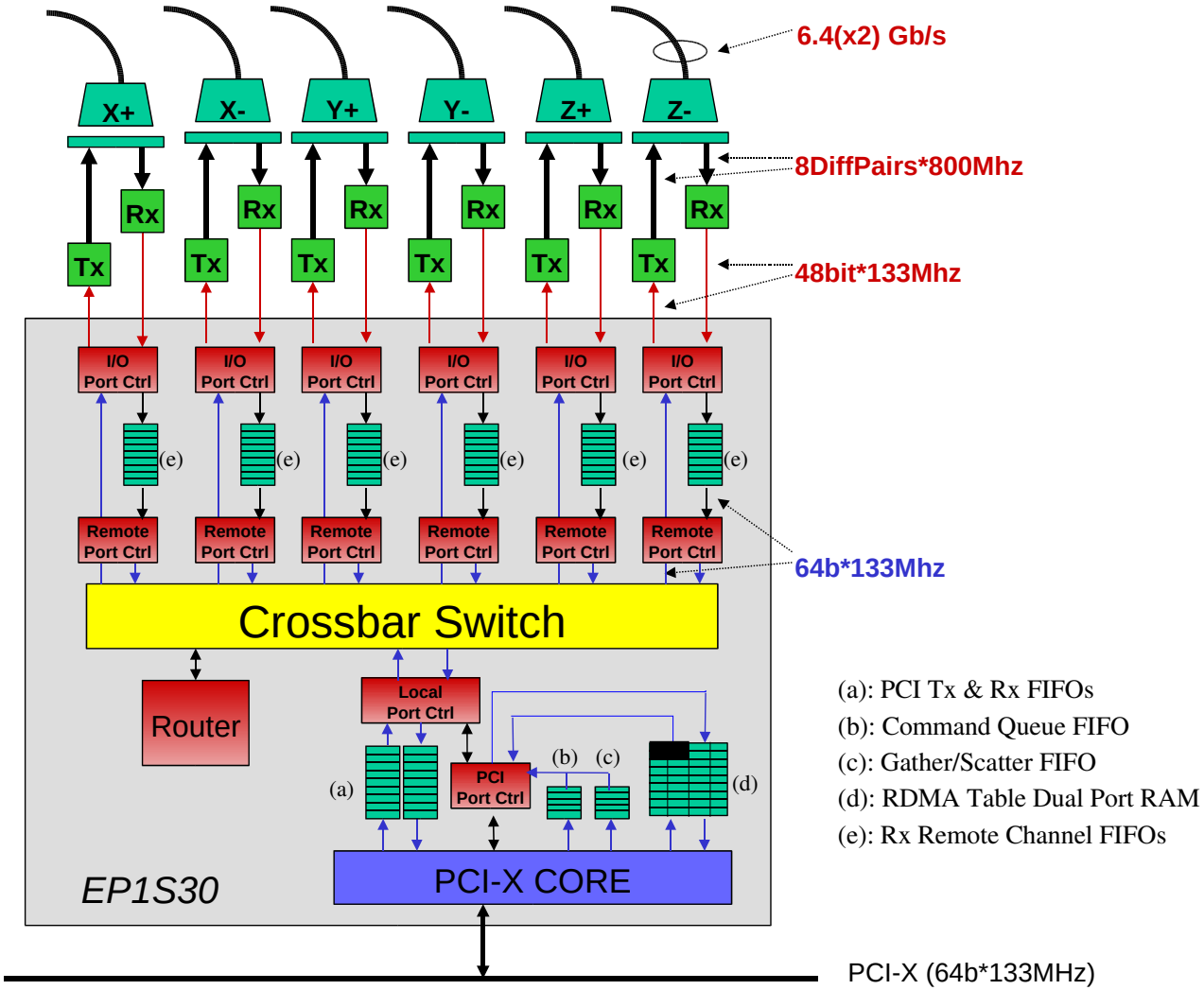
The interconnection card



- Altera Stratix EP1S30, 1020 pin package, fastest speed grade
- National Serializers/Deserializers DS90CR485/486, 48 bit 133 MHz

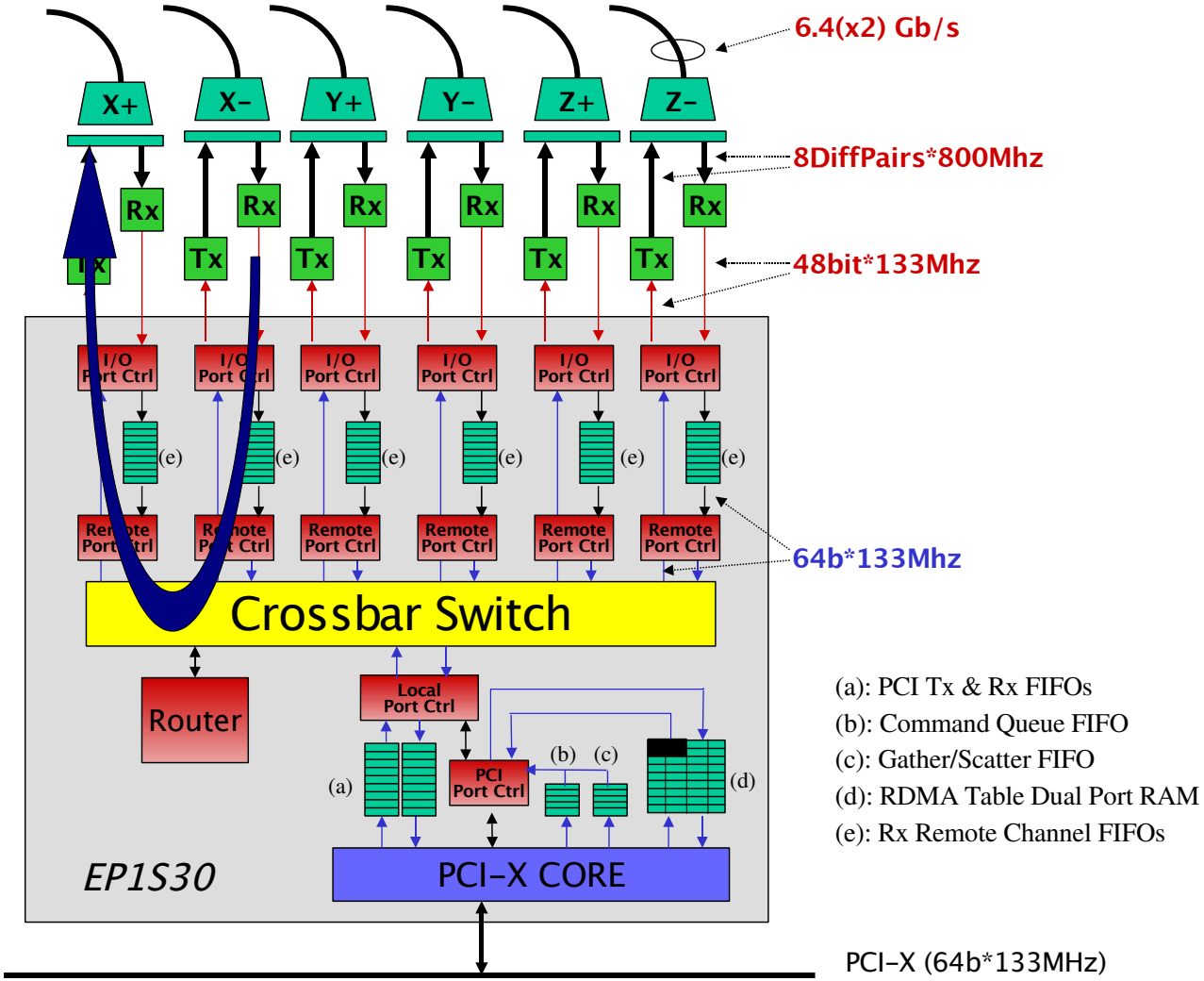
Usage of a programmable device allows possible logic redesign and quick on-field firmware upgrade.

Functional blocks



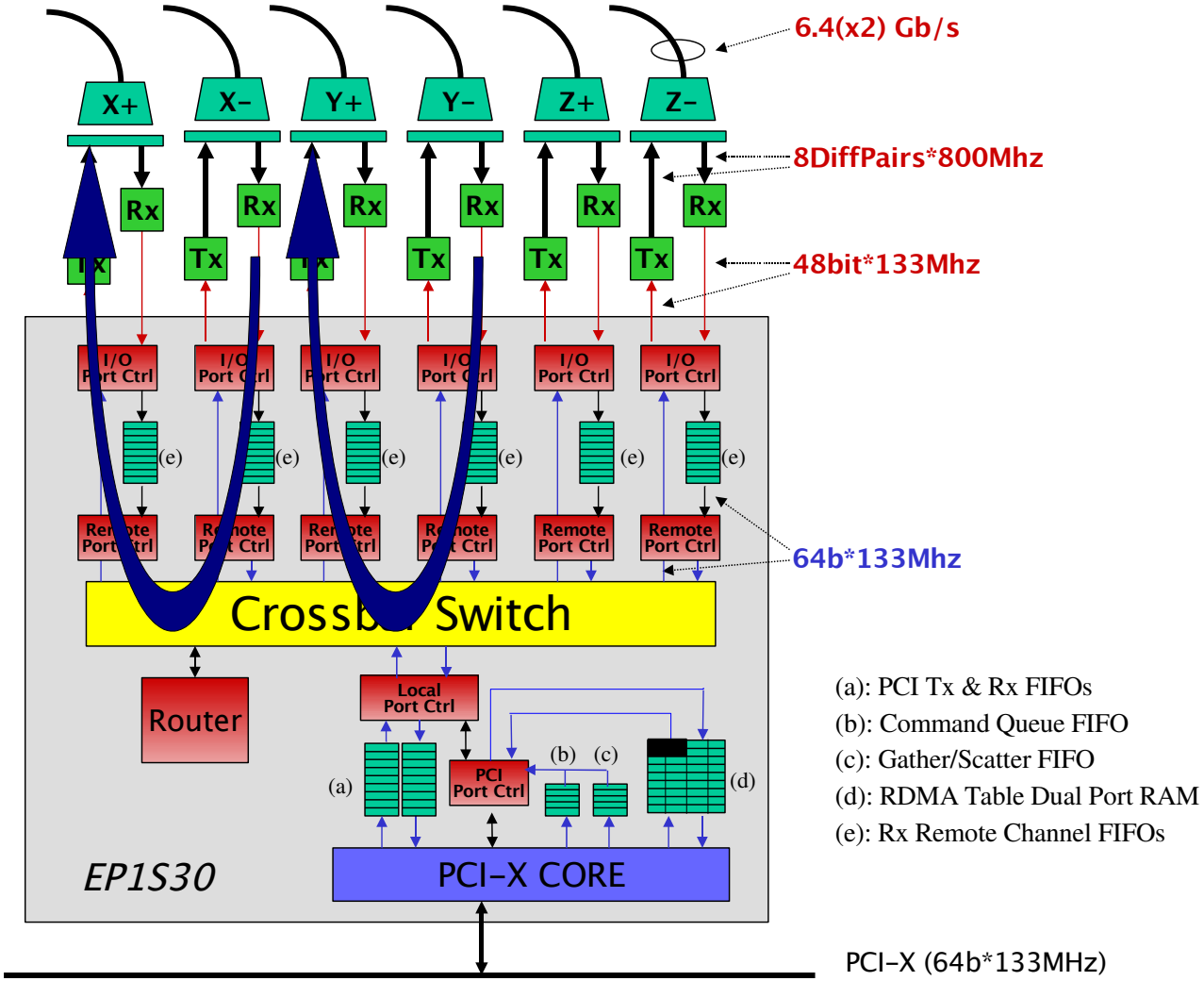
(a): PCI Tx & Rx FIFOs
(b): Command Queue FIFO
(c): Gather/Scatter FIFO
(d): RDMA Table Dual Port RAM
(e): Rx Remote Channel FIFOs

Functional blocks



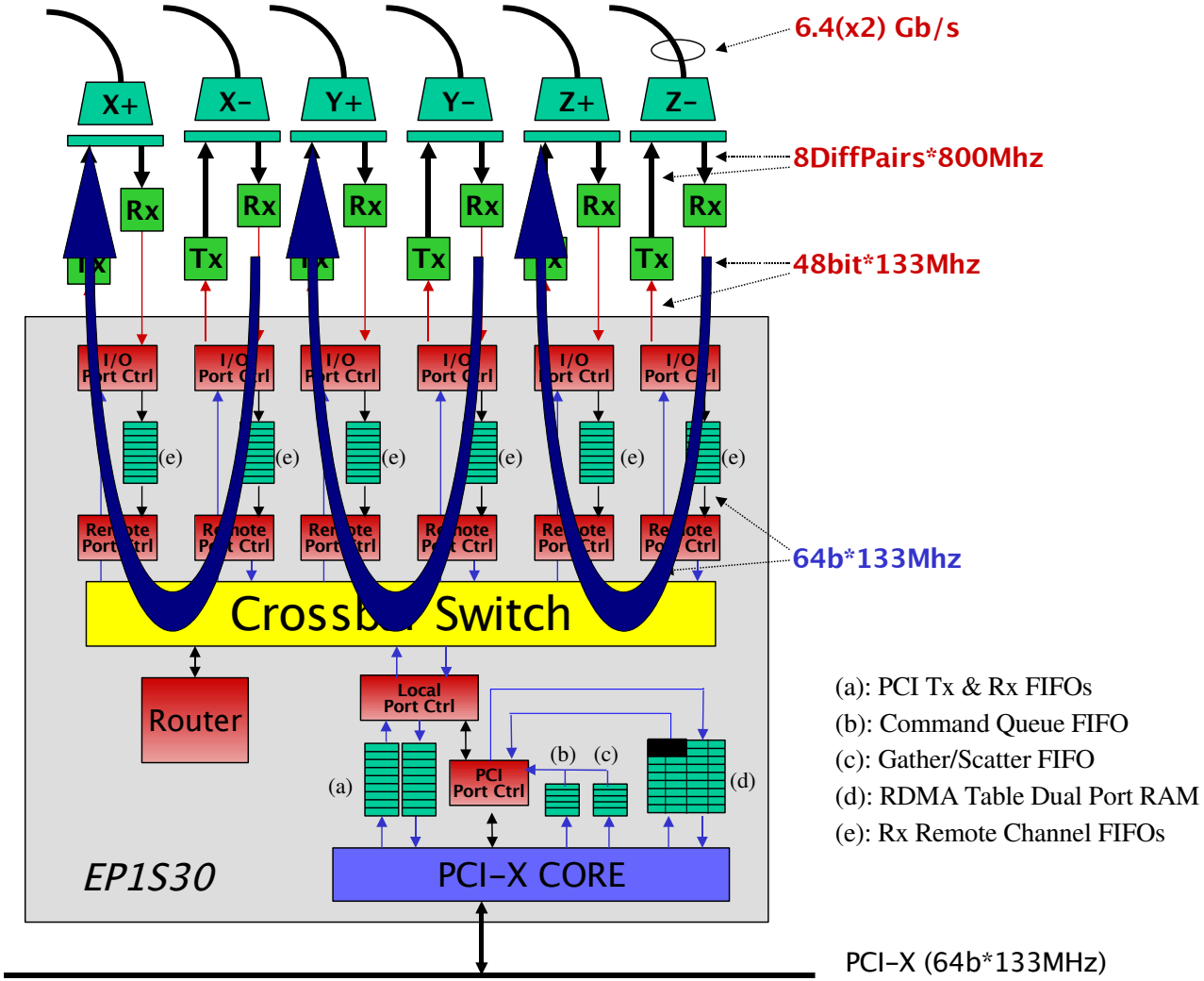
(a): PCI Tx & Rx FIFOs
(b): Command Queue FIFO
(c): Gather/Scatter FIFO
(d): RDMA Table Dual Port RAM
(e): Rx Remote Channel FIFOs

Functional blocks



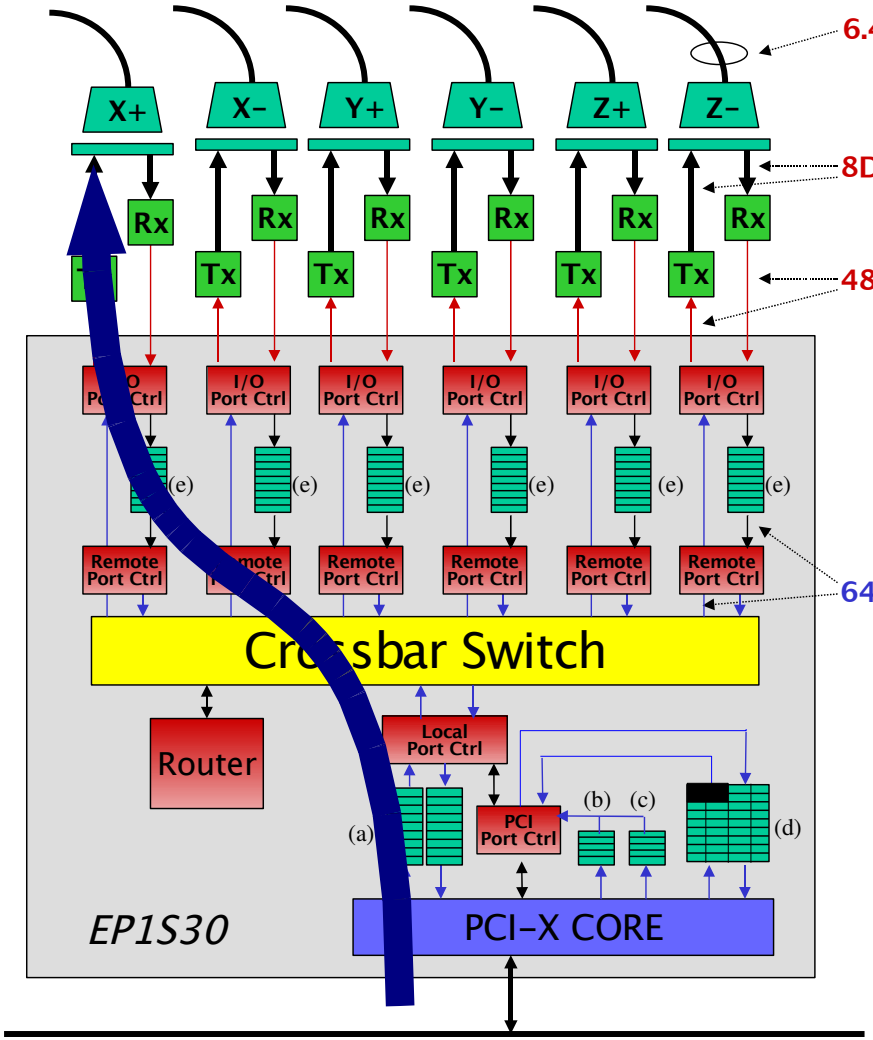
(a): PCI Tx & Rx FIFOs
(b): Command Queue FIFO
(c): Gather/Scatter FIFO
(d): RDMA Table Dual Port RAM
(e): Rx Remote Channel FIFOs

Functional blocks



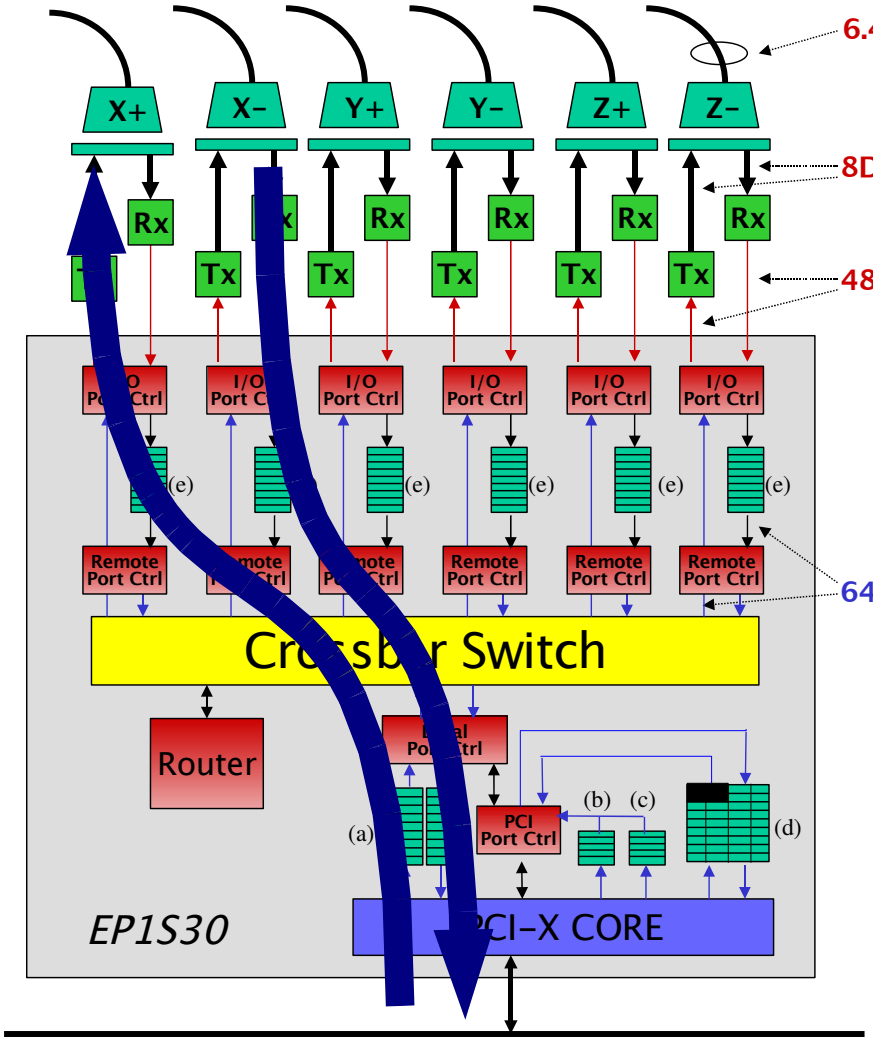
(a): PCI Tx & Rx FIFOs
(b): Command Queue FIFO
(c): Gather/Scatter FIFO
(d): RDMA Table Dual Port RAM
(e): Rx Remote Channel FIFOs

Functional blocks



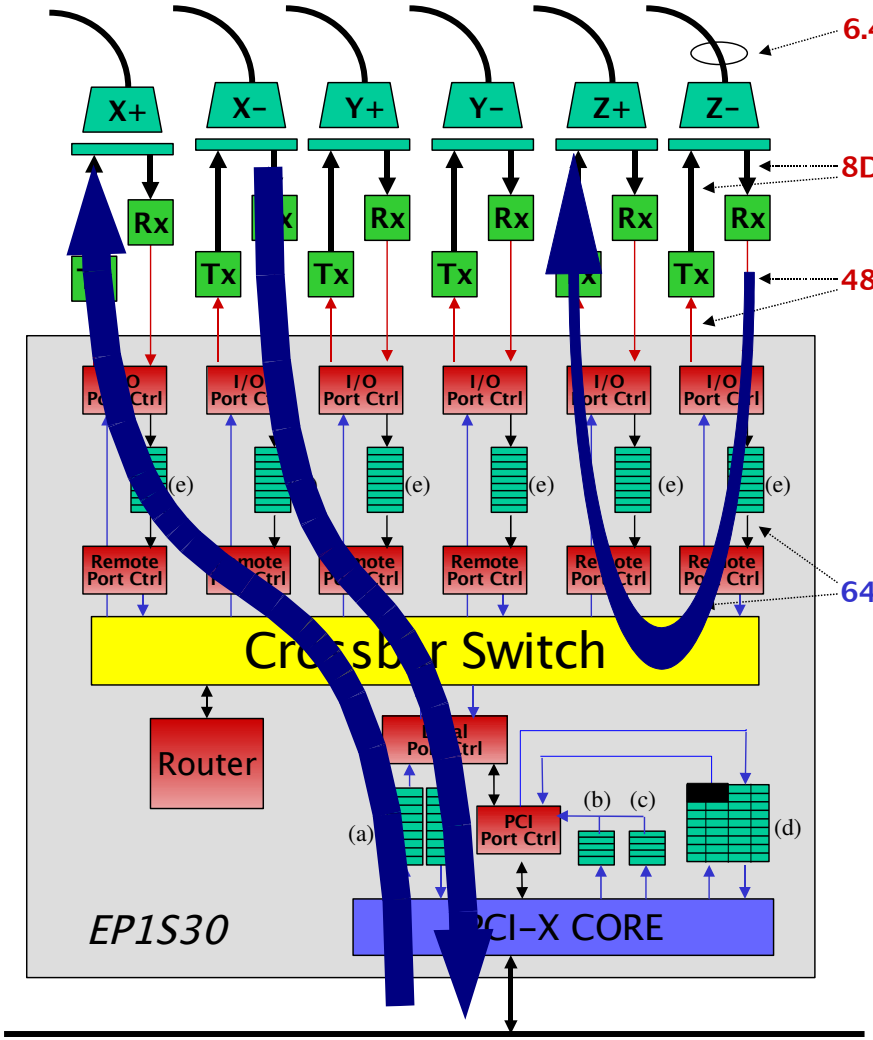
(a): PCI Tx & Rx FIFOs
(b): Command Queue FIFO
(c): Gather/Scatter FIFO
(d): RDMA Table Dual Port RAM
(e): Rx Remote Channel FIFOs

Functional blocks



(a): PCI Tx & Rx FIFOs
(b): Command Queue FIFO
(c): Gather/Scatter FIFO
(d): RDMA Table Dual Port RAM
(e): Rx Remote Channel FIFOs

Functional blocks



(a): PCI Tx & Rx FIFOs
(b): Command Queue FIFO
(c): Gather/Scatter FIFO
(d): RDMA Table Dual Port RAM
(e): Rx Remote Channel FIFOs

Software

A new piece of Hardware needs new Software

- Device driver
- Application level library
- Application level test suites
- MPI library
- MPI level test suites

Submitting job

A job submitting environment has been developed, aware of allowed network topologies on a given machine. A configuration file describes allowed topologies:

```
APE_NET_TOPOLOGY        4 4 8

APE_NET_PARTITION       full            4 4 8   0 0 0
APE_NET_PARTITION       single          1 1 1   0 0 0

APE_NET_PARTITION       zline0          1 1 8   0 0 0
APE_NET_PARTITION       zline1          1 1 8   0 1 0
```

Jobs are submitted with an mpirun derived script:

```
aperun -topo zline0 cpi
```

MPI Profiling

It can be very hard to produce optimized parallel code.

A tool for profiling MPI communication is under development, which can help understanding the communication requests of a certain code for a better tuning of its internal parameters.

Informations are gathered regarding:

- communication vs computation time
- time spent per communication function
- transferred data size per communication function
- transferred data size transfer per rank
- ...

Testbed

APE128

- 128 Dual Xeon "Nocona"
  - 3.4 *GHz*
  - 1 GB RAM DDR 333
- 6 TB NFS export
- Service 100 Mbit Network
- User access 1Gbit Network
- $4 \times 4 \times 8$ APENet network

Some notes about the assembling

To host the machine we needed to rework the infrastructure:

- floor
- air condition
- power supply with UPS

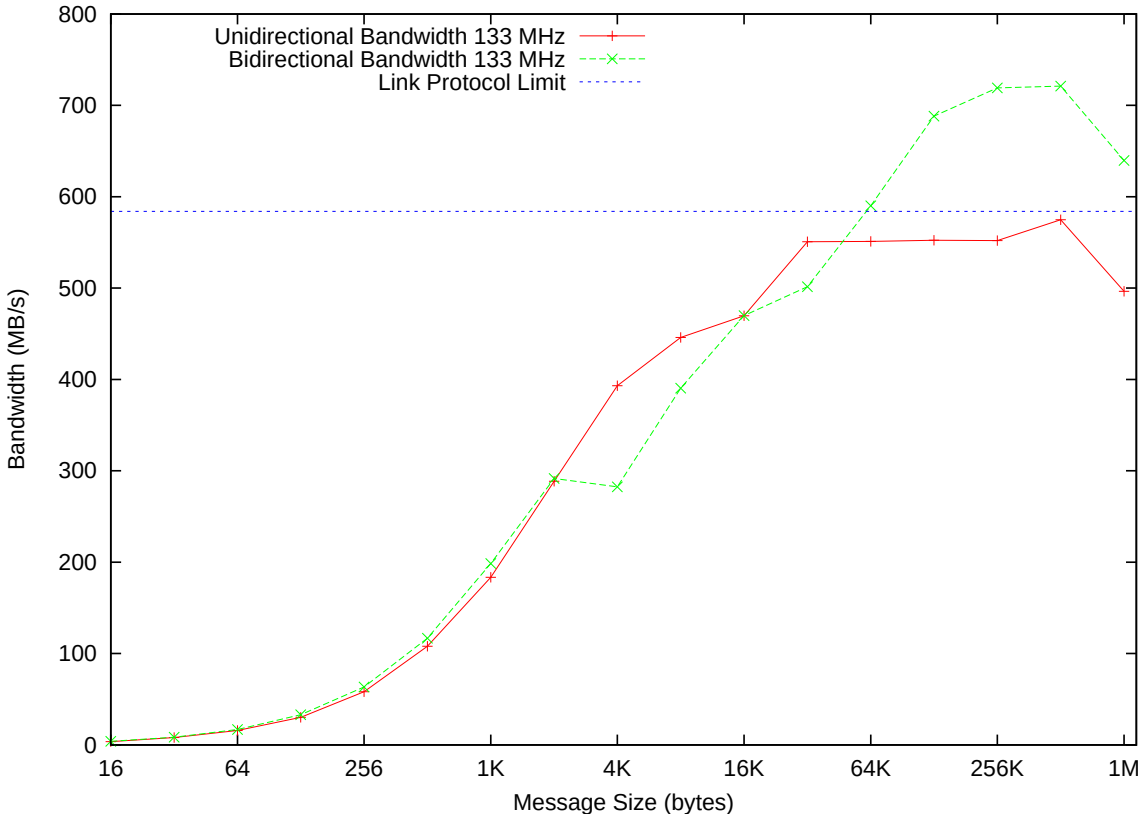
`APENet` assembling issues:

- 5% faulty devices
- 20% badly mounted devices
- 10% badly connected links

General issues:

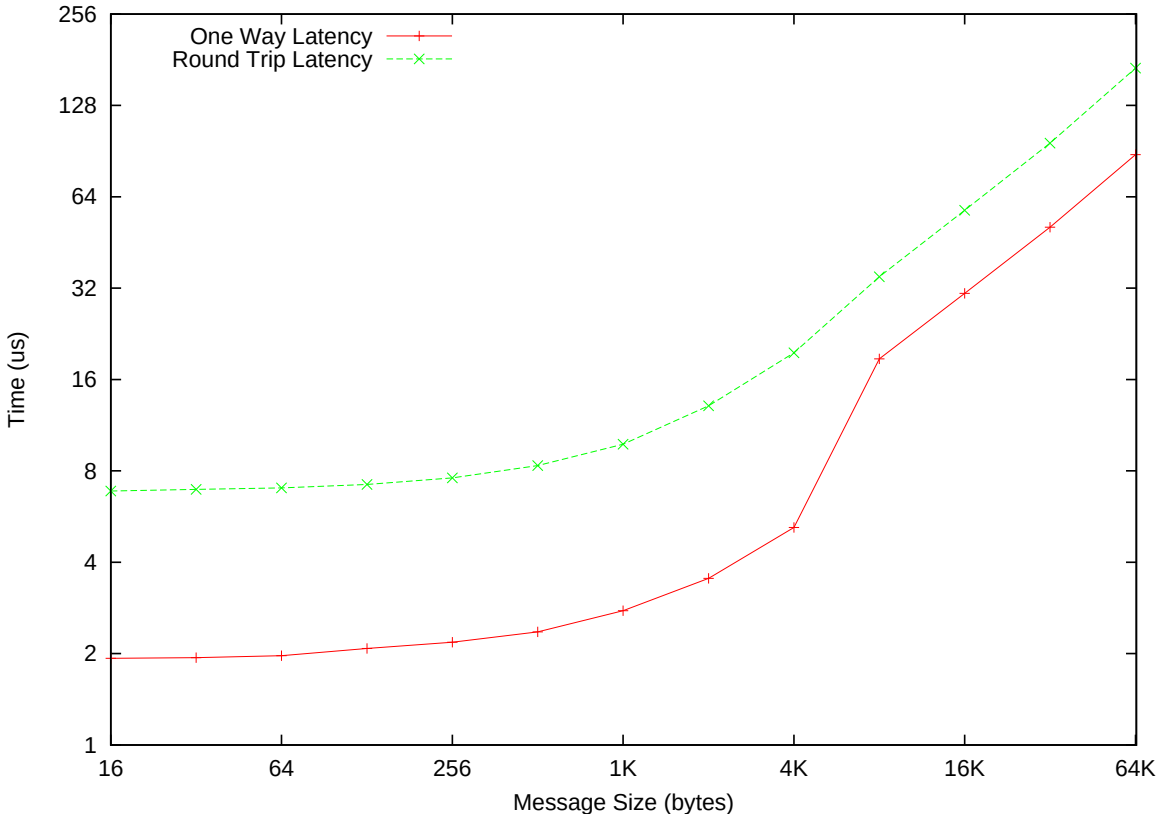
- 10% early dead disks

MPI Bandwidth benchmark



- Peak Unidirectional Bandwidth $\simeq 570 MB/s$
- Peak Bidirectional Bandwidth $\simeq 720 MB/s$
- Link physical limit $\simeq 590 MB/s$

MPI Latency benchmark



- Minimum Streaming Latency $\simeq 1.9\mu s$
- Minumum Round Trip Latency $\simeq 6.9\mu s$

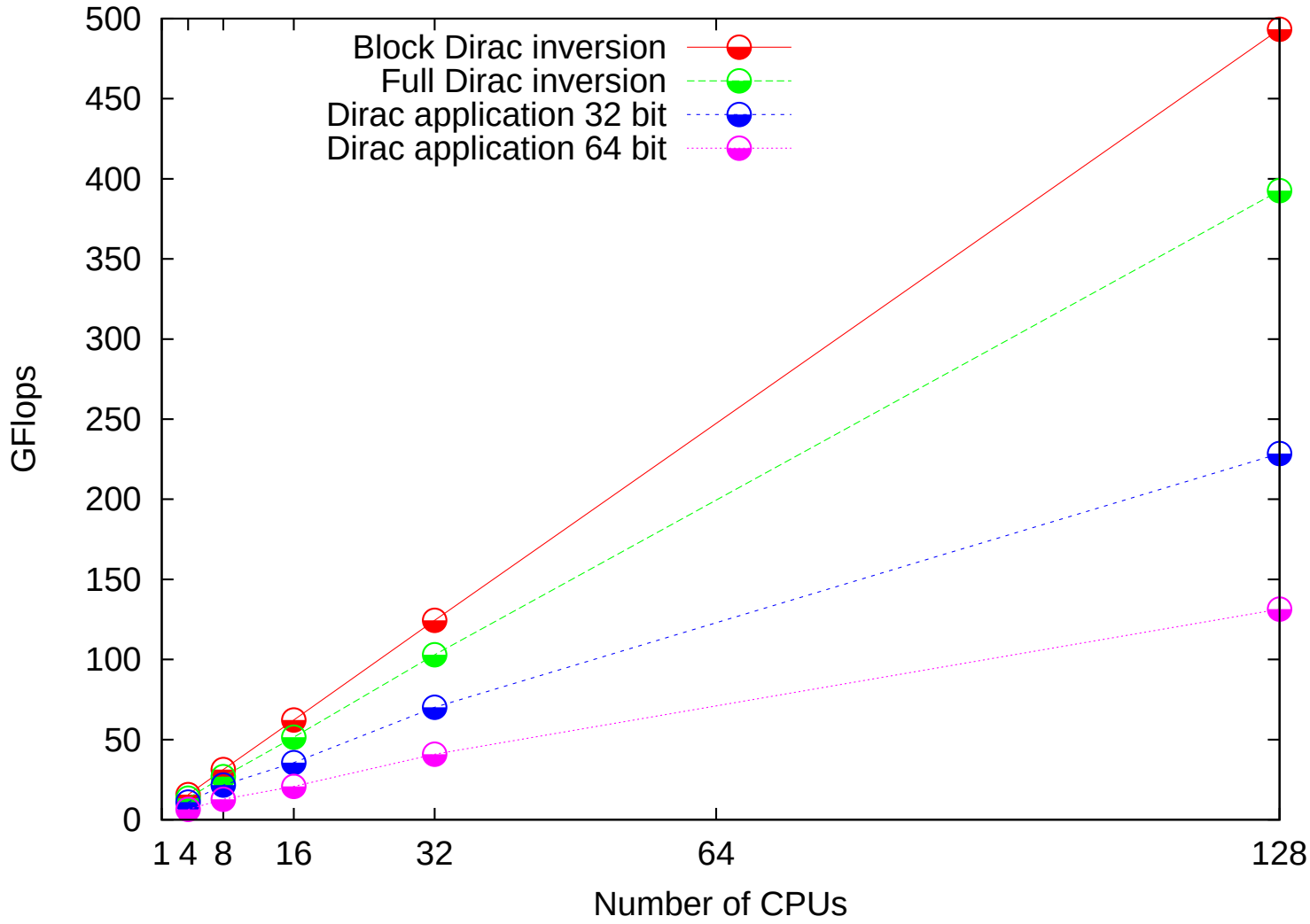
Real life benchmark

The performances (in floating point operation per second) of 4 LQCD core routines are shown. The tests are executed with a fixed local lattice size (8^4).
Allowed network topologies:

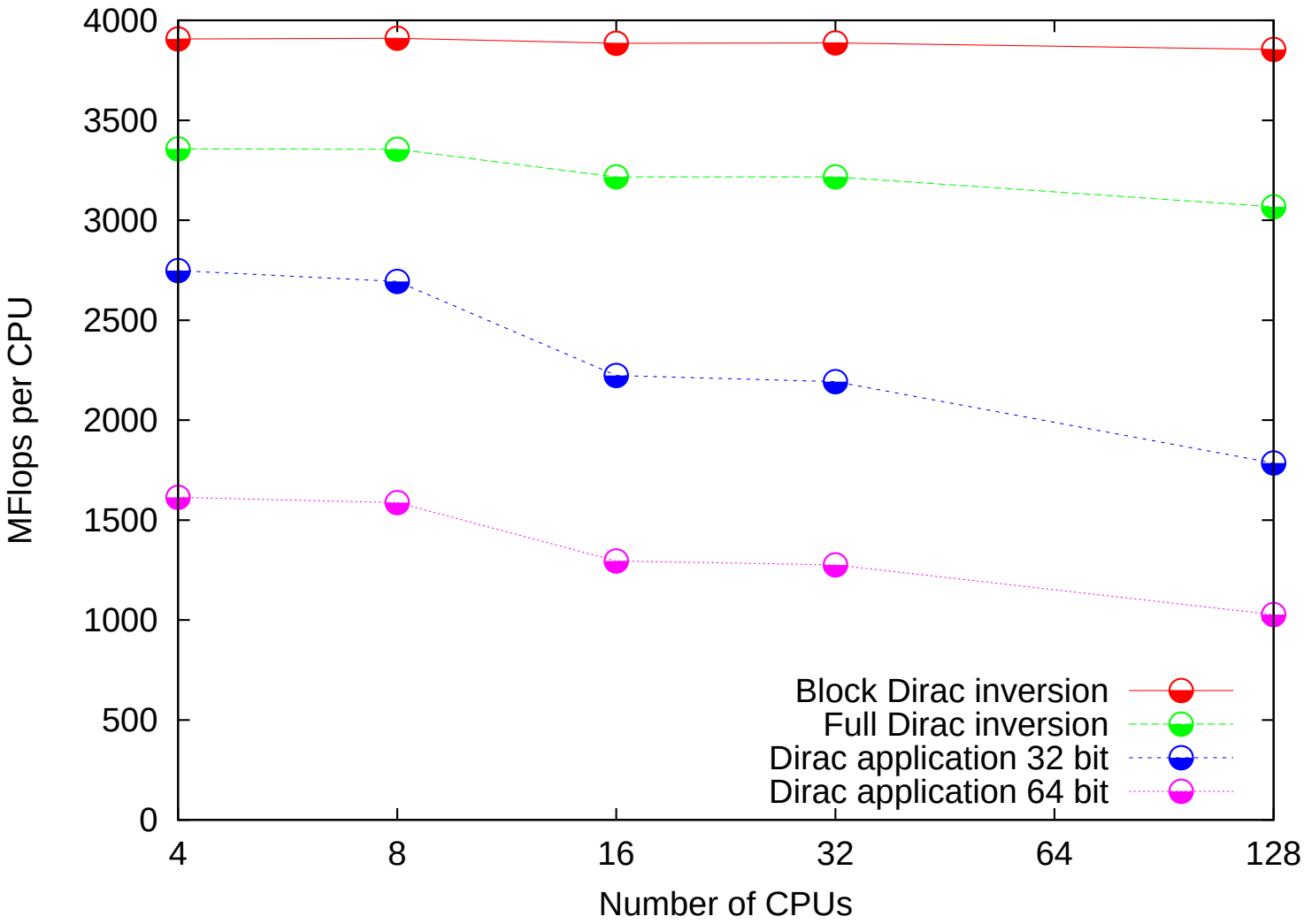
- 4 CPU: $4 \times 1 \times 1$
- 8 CPU: $1 \times 1 \times 8$
- 16 CPU: $4 \times 4 \times 1$
- 32 CPU: $1 \times 4 \times 8$
- 128 CPU: $4 \times 4 \times 8$

Best results are achieved when the global lattice geometry reflects the physical network topology.

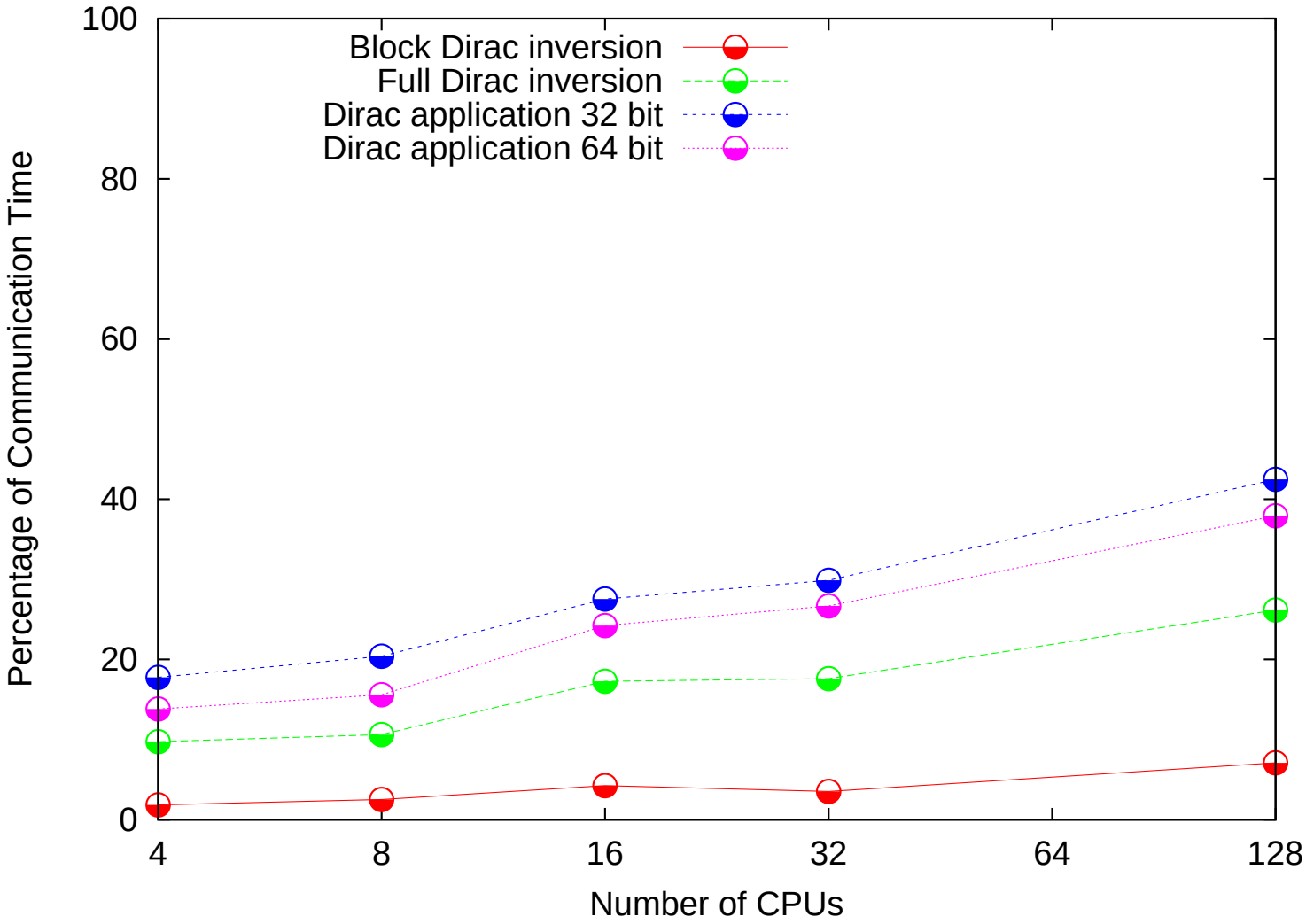
Application Scaling



Application Scaling



Application Scaling



Future work

Development is still in progress both for enhancing performances and for bug fixing.

Activity will be focused mainly on:

- Multiport introduction
- Reduce CPU usage for communication
- Reduce latency for medium-large size data transfers

Conclusions

Developing APENet has been a very interesting challenge and performances can be (still) compared with commercial interconnect systems in HPC.

Technology is running fast out there, and as we are not market driven (and under-sized), it's very easy to become obsolete.

The acquired know-how is going to be transferred in the next generation APE machines (going for PETAFlops?).

APENet people

- R. Ammendola
- R. Petronzio
- D. Rossetti
- A. Salamon
- N. Tantalo
- P. Vicini