Backend DAQ for sBelle Pixel Detector (an Update)

Jonas Hellhund, Andreas Kopp, Wolfgang Kühn, Johannes Lang,
Jens Sören Lange, Ming Liu, David Münchow, Johannes Roskoss,
Qiang Wang, (Tiago Perez, Daniel Kirschner)

II. Physikalisches Institut, Justus-Liebig-Universität Giessen

Colleagues involved in project,
but not (s)Belle members

Dapeng Jin, Lu Li, Zhen'An Liu, Yunpeng Lu,
Shujun Wei, Hao Xu, Dixin Zhao

IHEP Beijing

2nd International Workshop on DEPFET Detectors, Ringberg, May 3-6, 2009

PXD Frontend & Readout

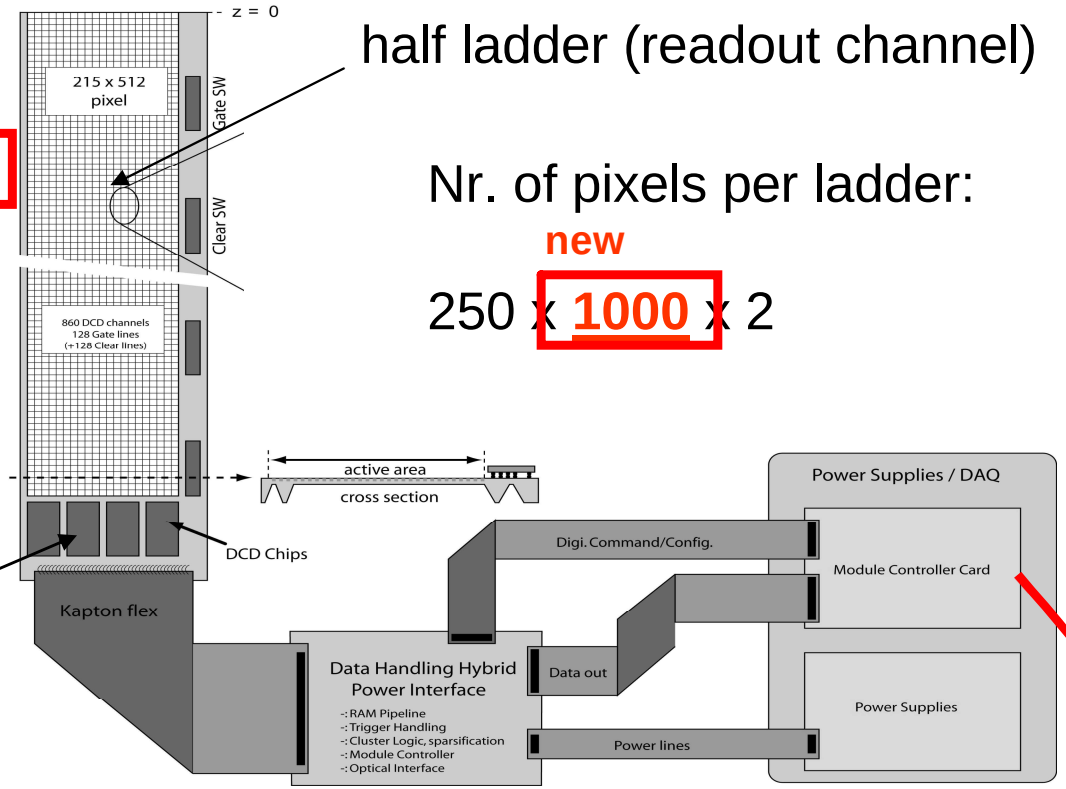
Total rate: ≤220 Gbits/s

Slide by Christian Kiesling

250

new

1000



half ladder (readout channel)

Nr. of pixels per ladder:

new

250 x 1000 x 2

- 44 half ladders:
  5.5 Million pixels (px)
- 1-2% occupancy:
- 55 kpx on at any time
- → about 10^5 px in each event
- Trigger rate: 10-30 KHz

  Total rate: 3 x 10^9 px/s
- 4 bytes per px (pos + ADC)

DHP:
zero-suppression
readout buffers (10 MByte)
slow control …

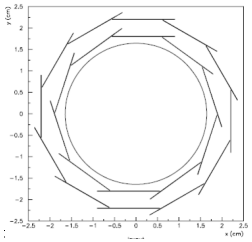
DHH: module controller
data link

Digitzation (DCD)

Backend DAQ
Subevent builder
Data reduction

≤5 Gbits/s
per R/O channel

Data Rates of sBelle Vertex Detectors (conservative estimate)

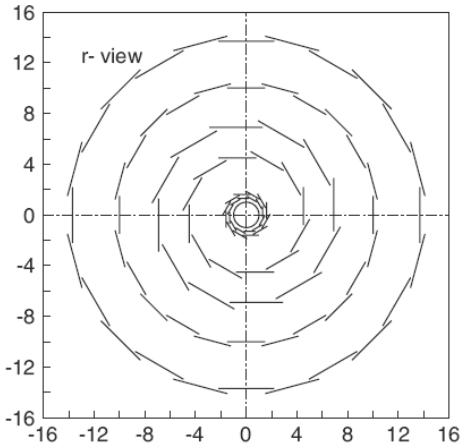


PXD raw data = 5 Gbit/s x 44 links = 27500 Mbyte/s

assume 30 kHz and new z segmentation (Valencia proposal)
assume 2-hit clusters, not 3-hit clusters

assume hit counting, not cluster counting

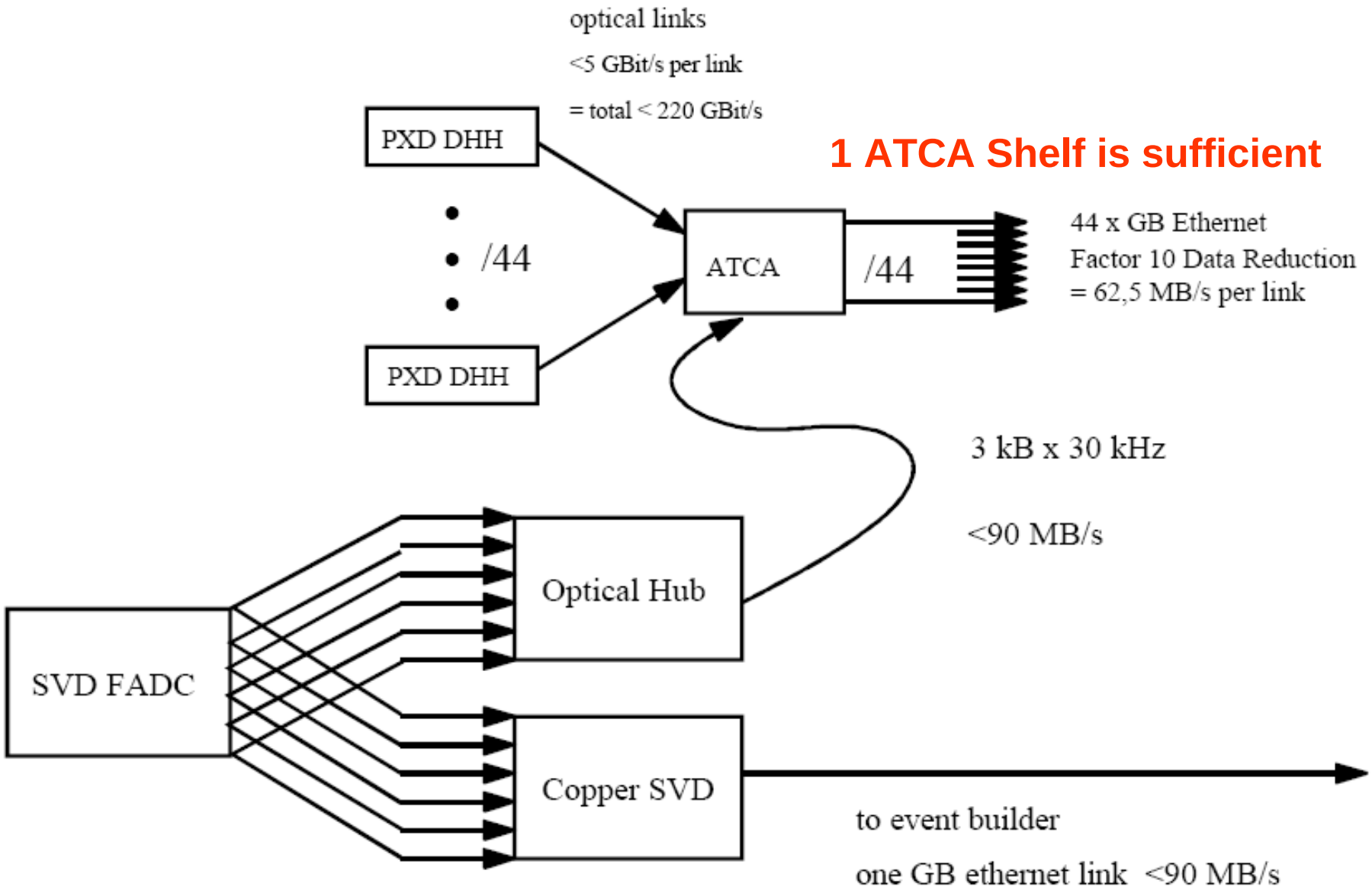
~ 7 DVDs per 1 second



SVD raw data = 3 kByte x 30 kHz = 90 Mbyte/s

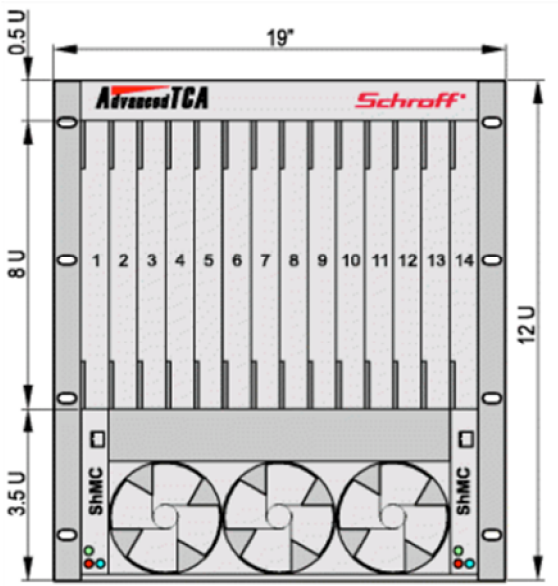
new radius of 1st layer will be at radius of current 2nd layer
$\rightarrow$ occupancy $\leq$ 10% expected
shorter shaping time of APV reduces factor 12.5
hit time reconstruction reduces factor 8

PXD Readout Scheme, 2nd Idea

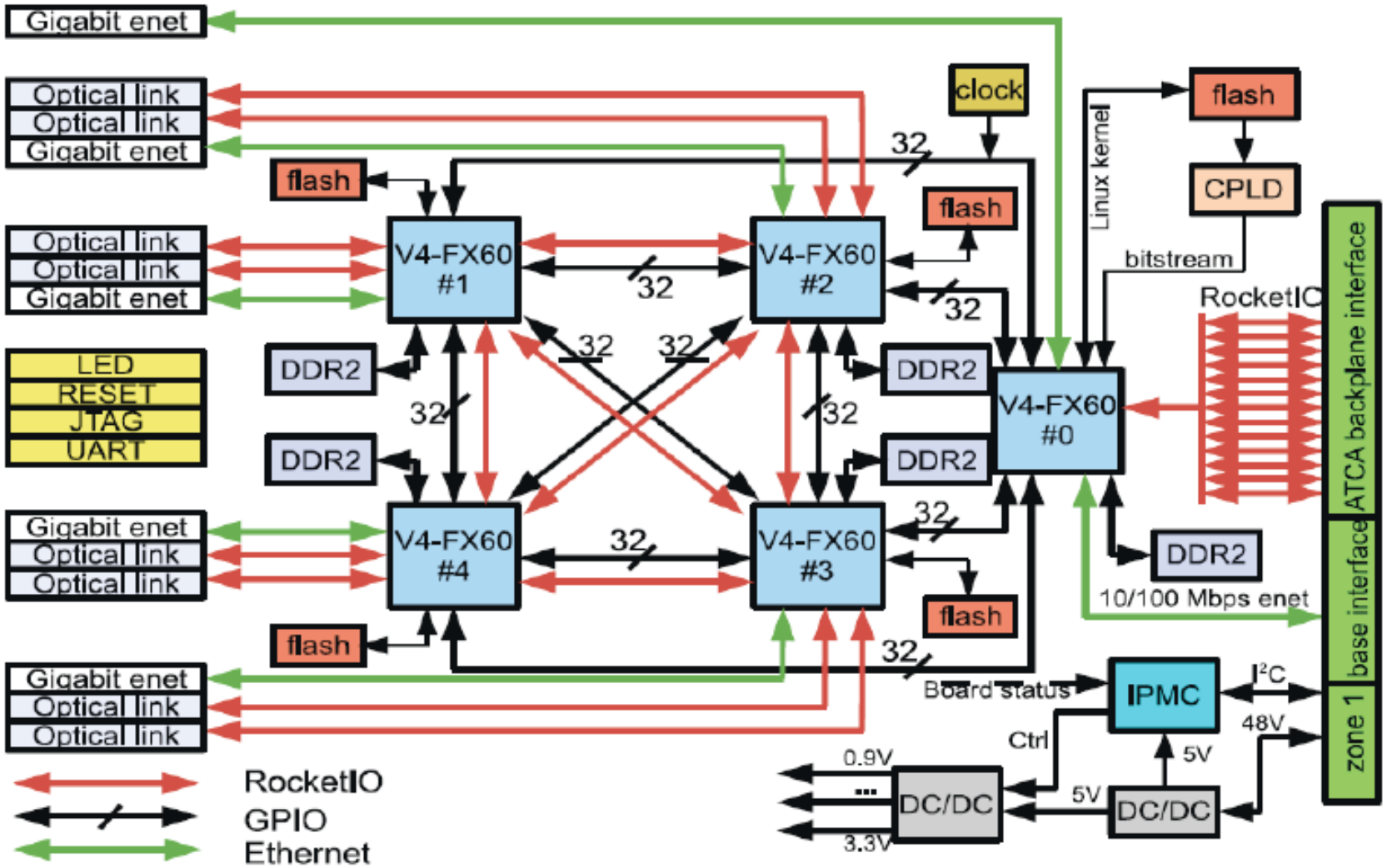


Compute Node (CN) Concept

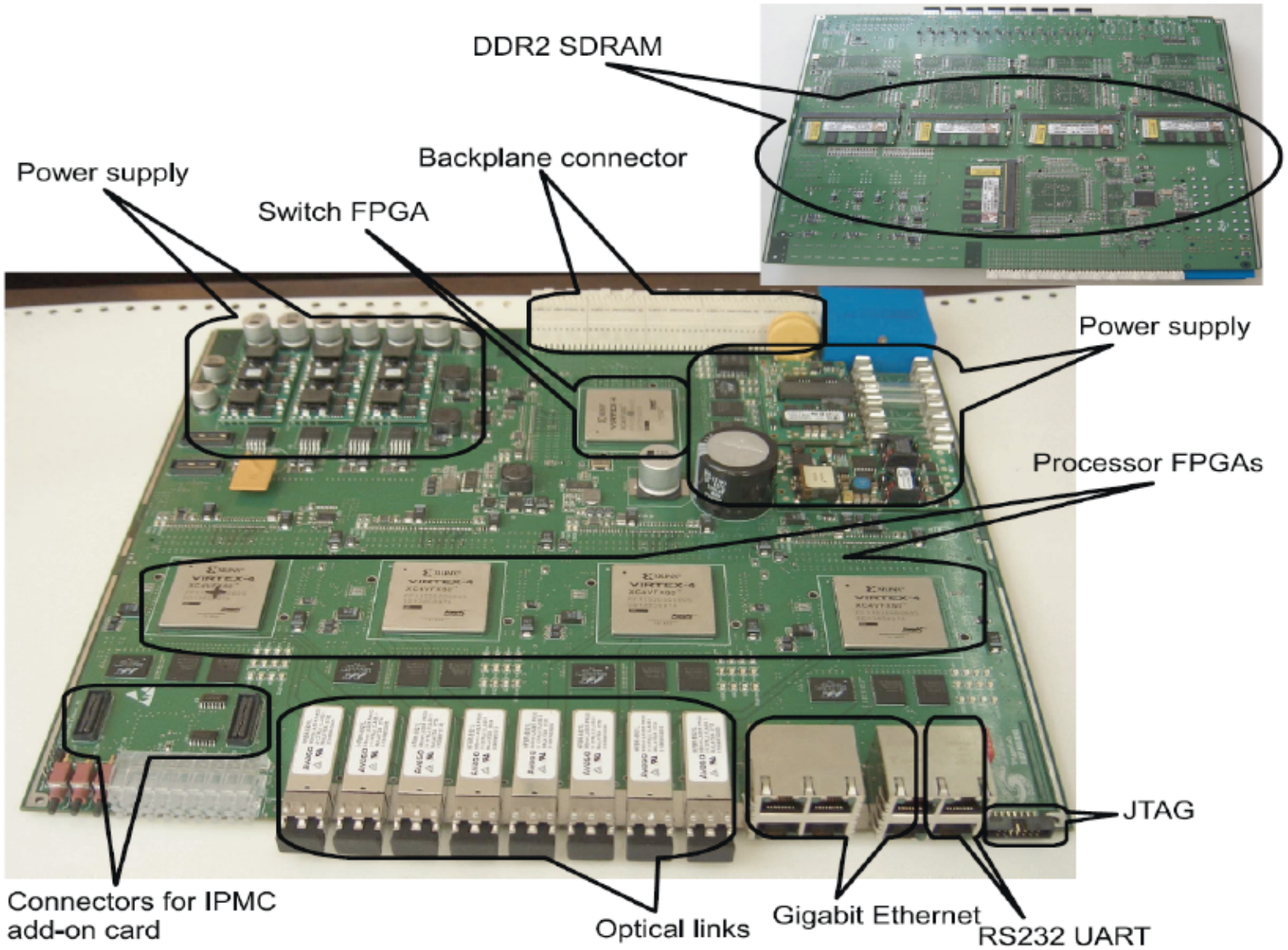
- **5 x VIRTEX4 FX-60 FPGAs**
  - **each FPGA has 2 x 300 MHz PowerPC**
  - **Linux 2.6.27 (open source version), stored in FLASH memory**
  - **algorithm programming in VHDL (XILINX ISE 10.1)**
- **ATCA (Advanced Telecommunications Computing Architecture) with full mesh backplane point-to-point connections on backplane from each CN to each other CN, i.e. no bus arbitration**
- **optical links (connected to RocketIO at FPGA)**
- **Gigabit Ethernet**
- **ATCA management (IPMI) by add-on card**



Compute Node Architecture



Compute Node (CN) Version 1.0

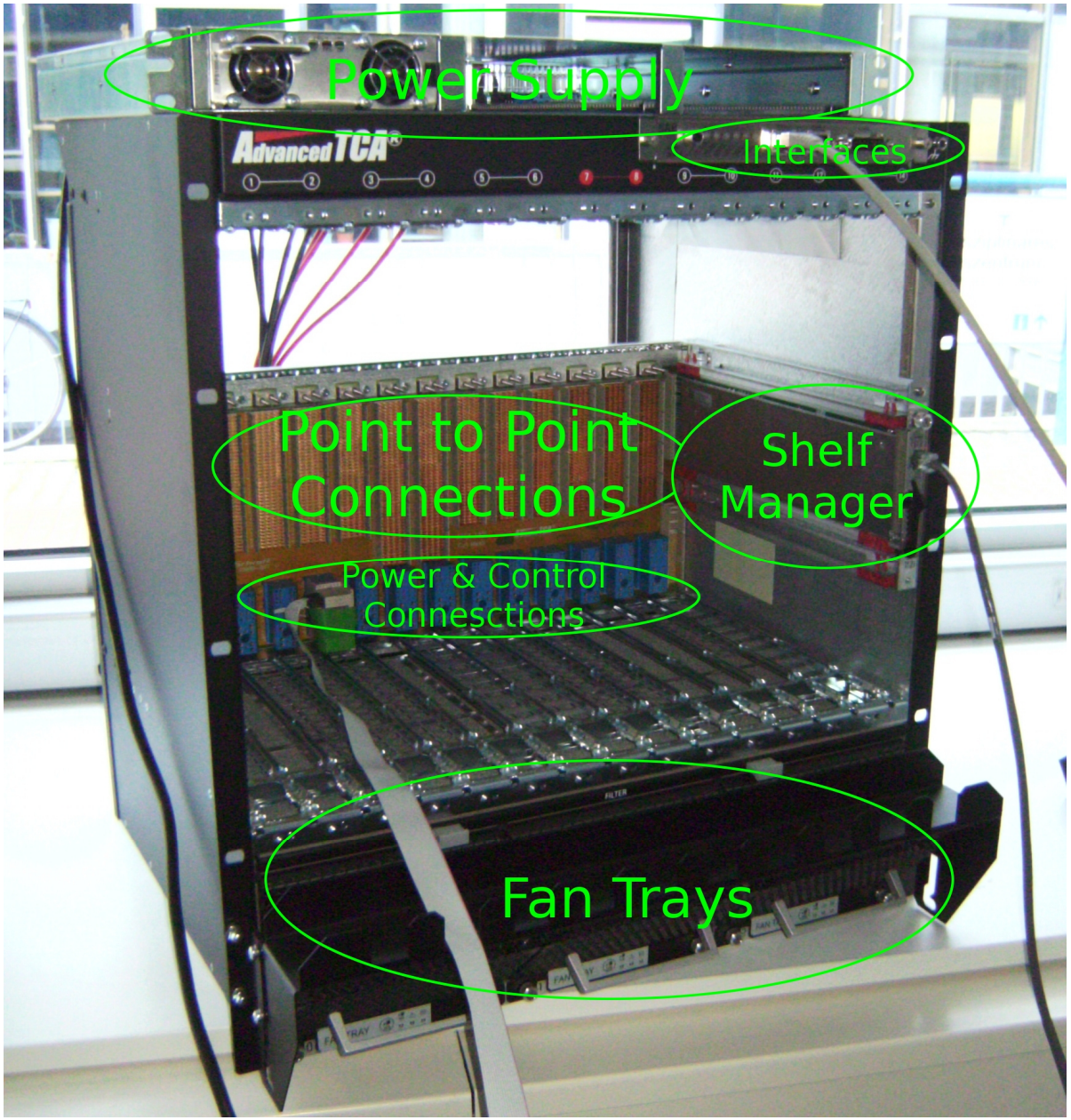


Compute Node Data Transfer

- **All 5 FPGAs are connected pairwise (on the board) by**
  - **one 32-bit general purpose bus (GPIO)**
  - **one full duplex RocketIO link**
- **4 of 5 FPGAs have two RocketIO links routed to front panel using Multi-Gigabit Receivers (MGT)**
  → optical links
- **1 of 5 FPGAs serves as a switch**
  → has 13 RocketIO links to all the other compute nodes in the same ATCA shelf
- **All 5 FPGA have a Gigabit Ethernet Link routed to front panel**

total integrated bandwidth • 32 Gbit/s
(all channels summed, theoretical limit)
per 1 board

ATCA Shelf



1 kW

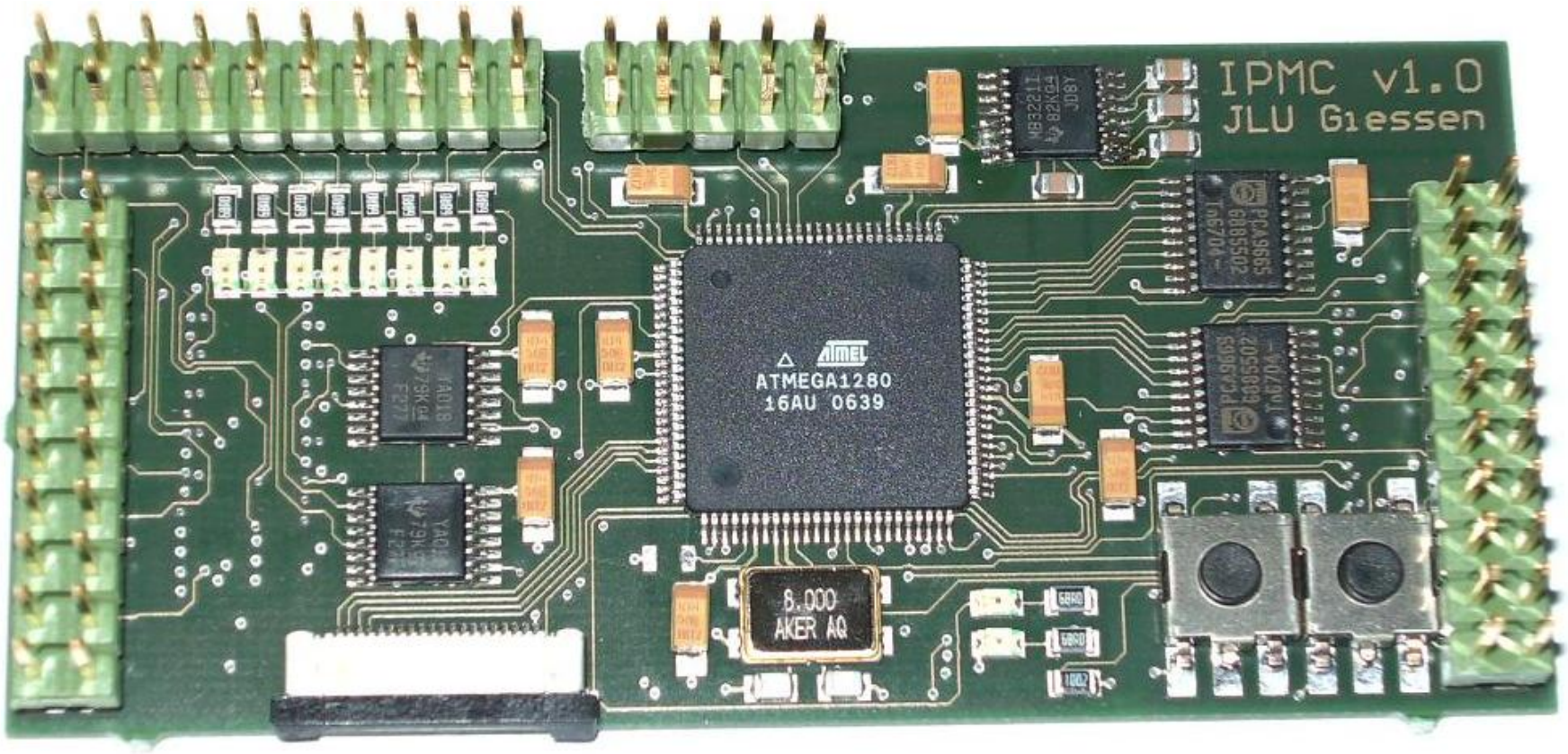
Test Results of CN Version 1.0

- optical links ☑
  @ 1.6 Gbps to TRB2 (HADES TPC board,
  CERN HPTPC and ETRAX 1-chip PC),
  0 bit error for 150-hour test
- Gigabit Ethernet ☑
  - 0.3 Gbit/s TCP
  - 0.4 Gbit/s UDP
- JTAG chain ☑
- Boot up sequence of all 5 FPGAs ☑
- CPLD+Flash system start-up mechanism ☑
  and remote reconfigurability
- DDR2 SDRAM ☑
  (multi-port read/write tested)
- other peripherals tested

IPMI

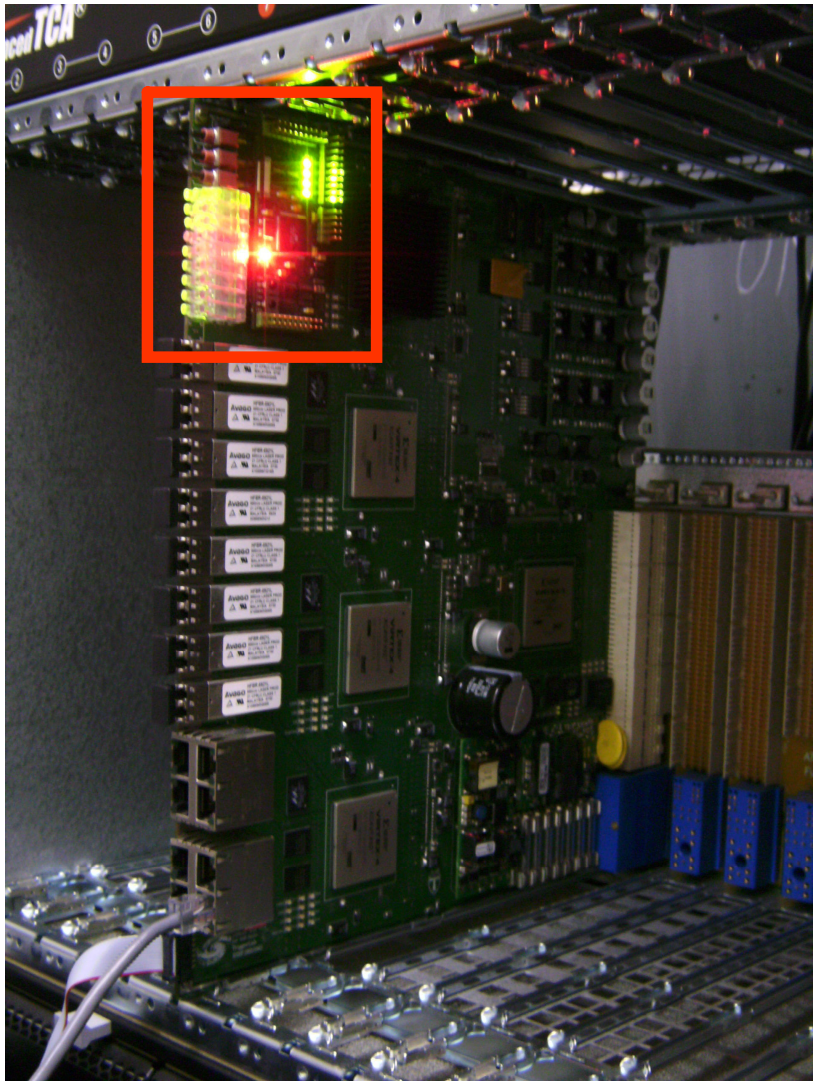
- Intelligent Platform Management Interface
- I^2C
  2-line serial interface (clock, data)
- ATCA Power Management
  ~180 W / compute node needed
  but only 10 W management power provided @ power-up
  → request to shelf manager
- CN piggy-back add-on card 75x35 mm
  design/layout in Giessen
  Microcontroller AVR Atmega 1280
  2 x 60 pin connector to CN
- Additional tasks:
  read temperature,
  read voltages (0.9/1.2/1.8/3.3/5.0 V, ADC via I^2C),
  allows for remote reset/reboot,
  hot swap
  (i.e. communicate to shelf manager
  „I will be disconnected from the backplane now“)

IPMI Add-On Board



Johannes Lang

IPMI Add-On Board

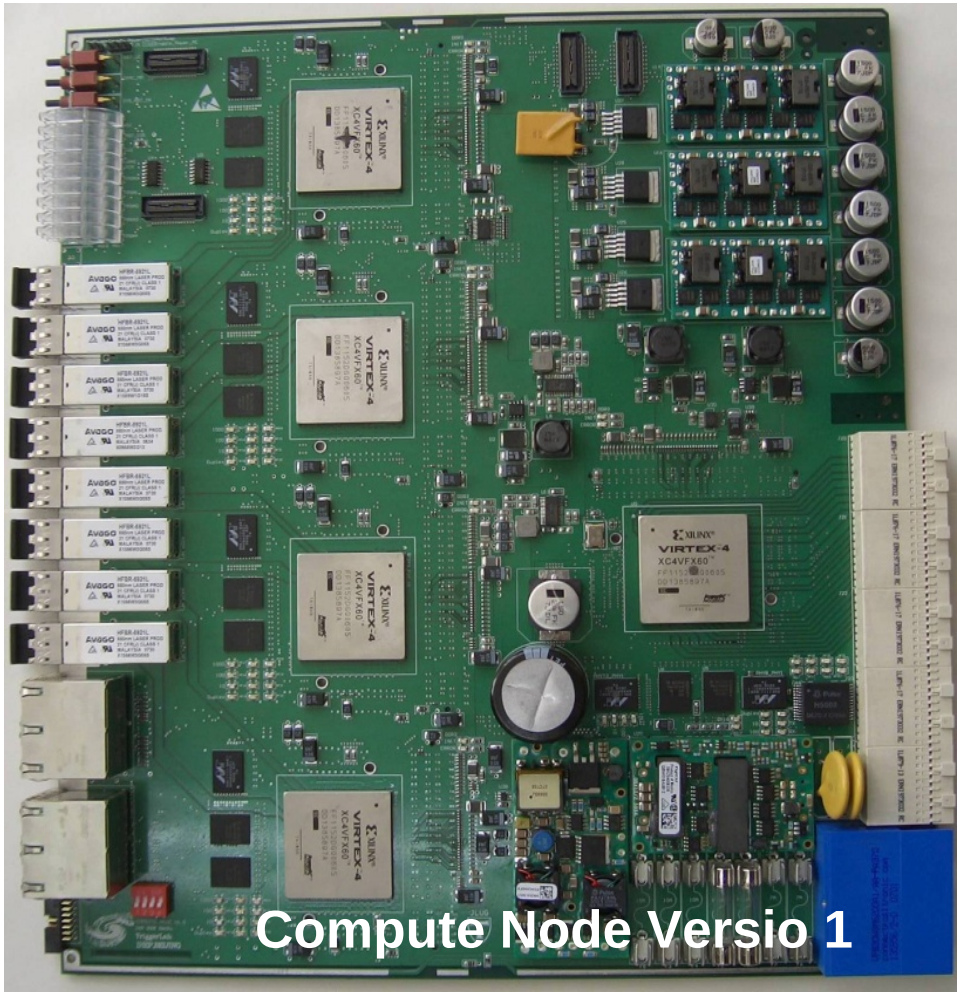


Johannes Lang

Debugging of Compute Node Version 1

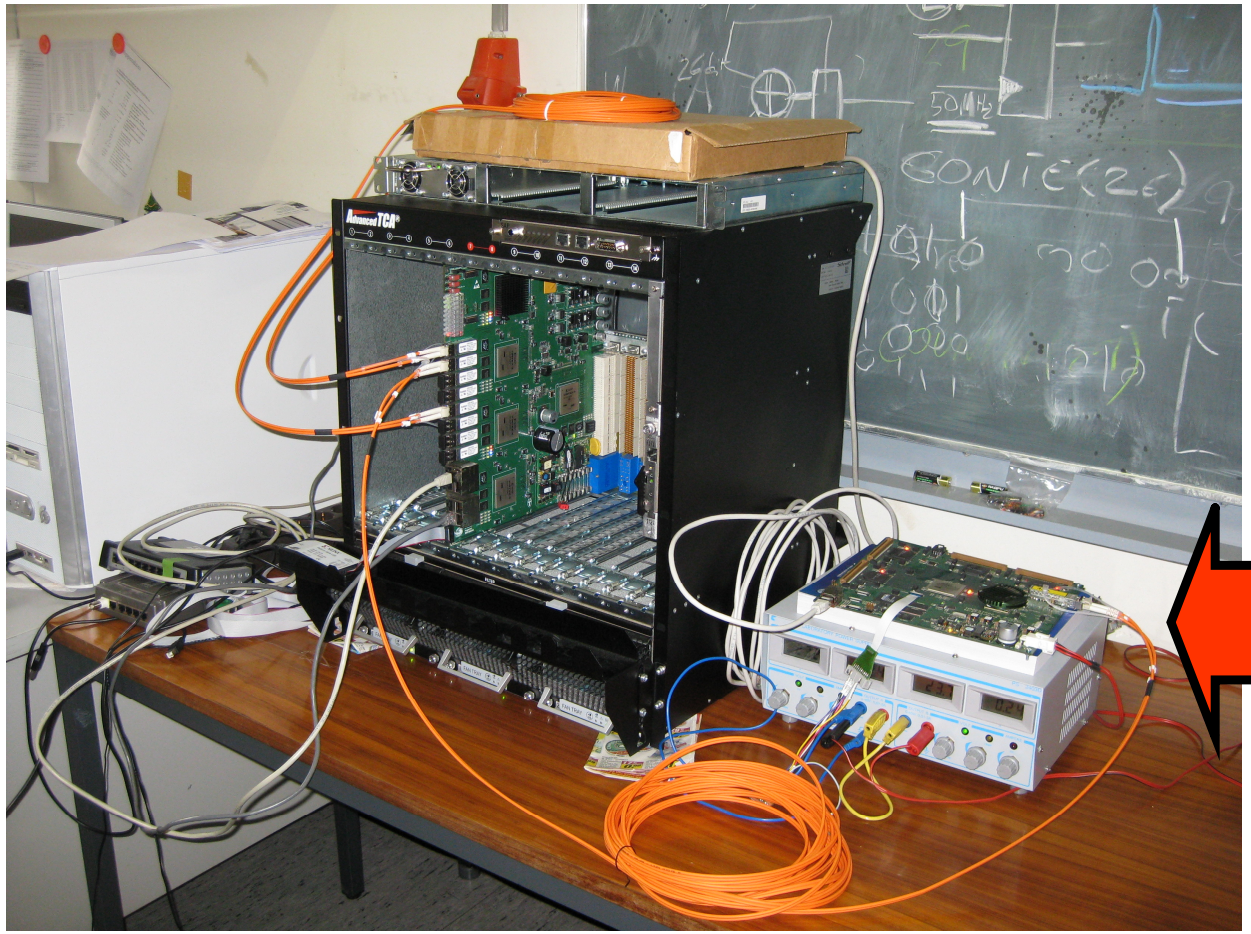
some problems identified **(Qiang Wang, Ming Liu, IHEP Group)**

- few oscillators wrong frequency
- PHY chip (88E1111) had pin connection which had to be re-mapped by Linux kernel
- FPGA0 Ethernet signal layout length exceeded the design rule
- problem with I^2C chip address setting (for temperature sensors)
- profile of ethernet plugs > ATCA 1 slot width
- RocketIO not AC coupled to backplane $\rightarrow$ capacitors added
- FPGA0 ethernet base channel was connected to backplane in 100 Mbps mode $\rightarrow$ now GBit
- and some additional small errors
- Note: this is a **14-layer PCB**



Demo System For REALTIME'09 in Beijing

Optical link data transfer at 1.6 Gbit/s



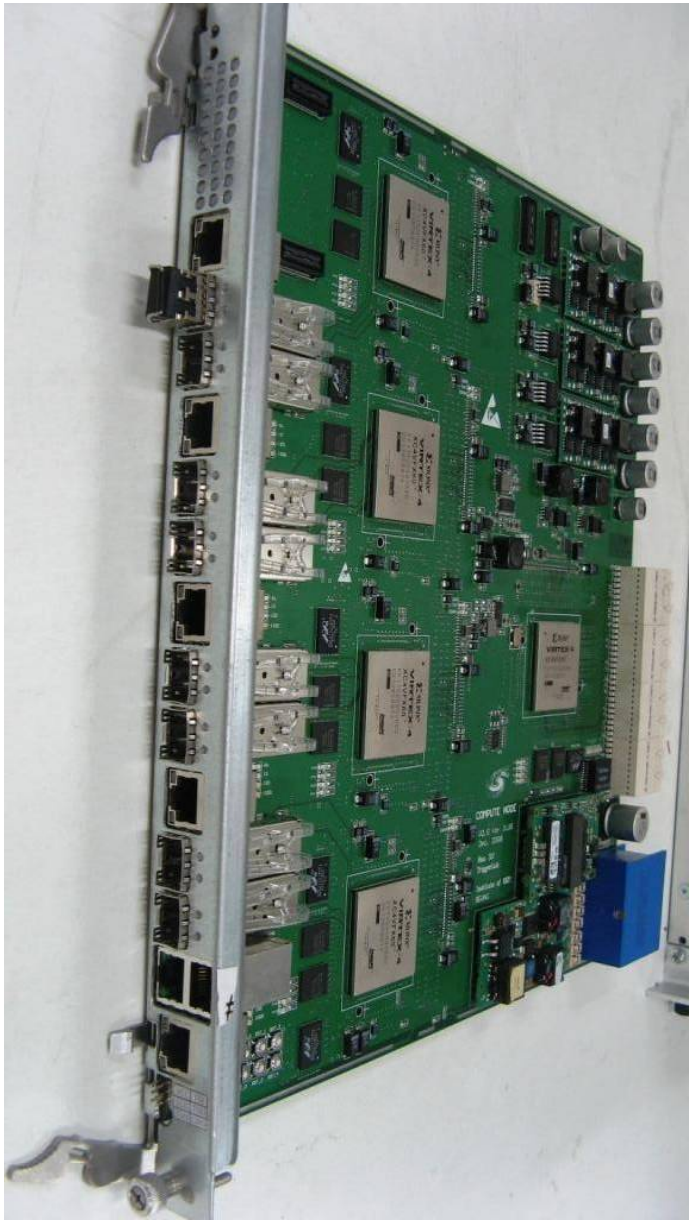
TRB
= Hades TDC Board
(CERN HPTDC)

Update on the Compute Node: Version 2

- **Bugs fixed**
- **arrived at IHEP on April 20**
- **maybe next month:**
  4 boards for Giessen
  (FPGA‘s have to be ordered first)
- **One of the boards for Giessen will have**
  - **new optical transceiver**
  - **new FPGA**

  for $\leq$ 5 Gbit/s optical data transfer

  (now: $\leq$ 2 Gbit/s)



Dapeng Jin, Lu Li, Zhen'An Liu, Yunpeng Lu, Shujun Wei, Hao Xu, Dixin Zhao (IHEP Beijing)

FPGA Algorithms

Panda Track Finder and Track Fitter for FPGA

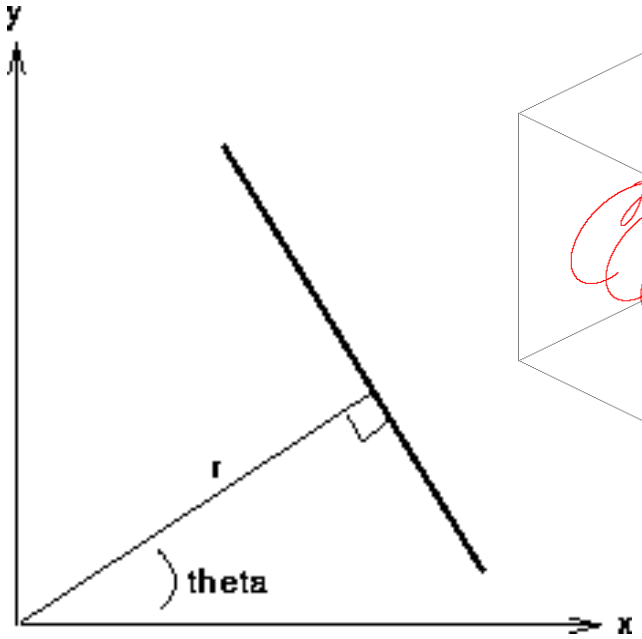
- **Track Finder for PANDA Straw Tube Tracker + SVD helix, 30+(5-7) hits per 1 track, ~10 tracks per event**
- **Field B_z=2 T (TOSCA Field Maps)**
- **Step #1: Conformal Map**
- **Step #2: Hough Transform**
- **Example: 10 Muon Tracks p=1 GeV/c**

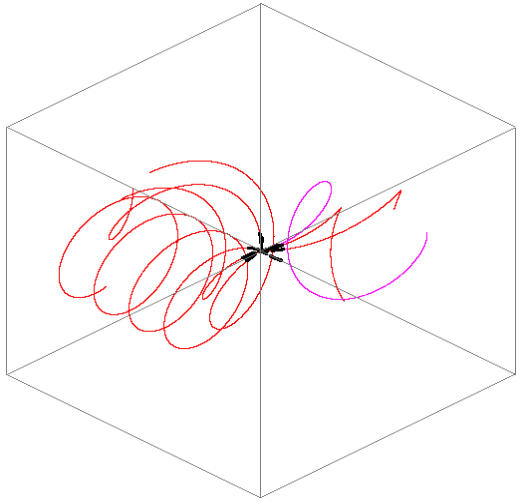
David Münchow

$$x' = \frac{x - x_0}{r^2}$$

$$y' = \frac{y - y_0}{r^2}$$

$$r^2 = (x - x_0)^2 + (y - y_0)^2$$





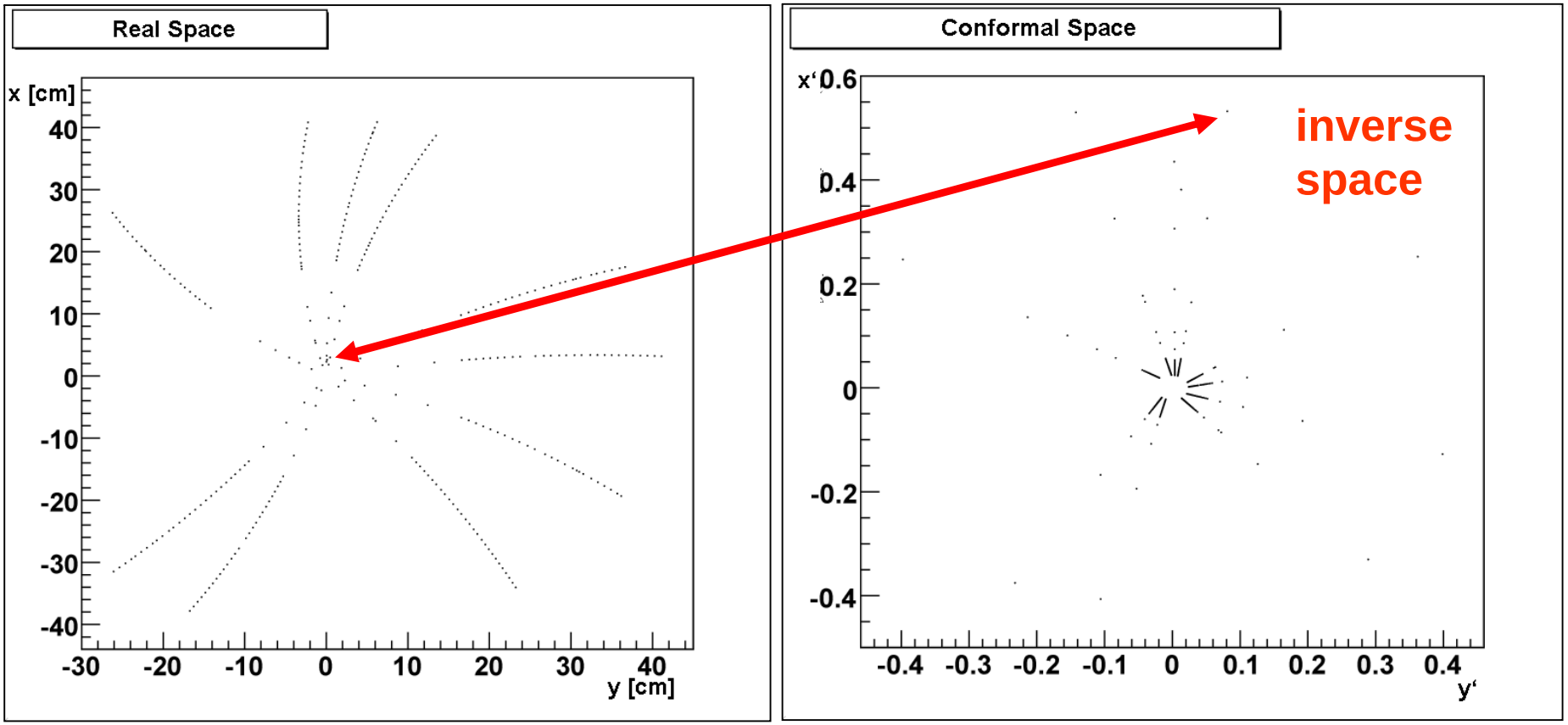
$$r(\theta) = x\cos\theta + y\sin\theta$$

Conformal Map Track Finder (not Track Fitter)

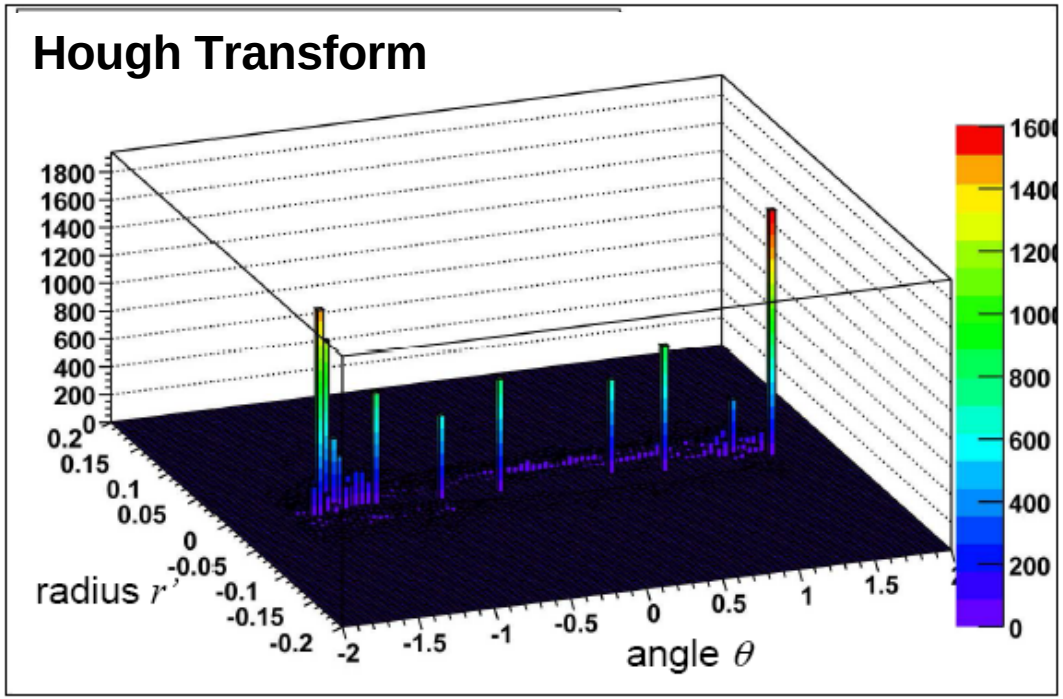
David Münchow

real space

conformal space

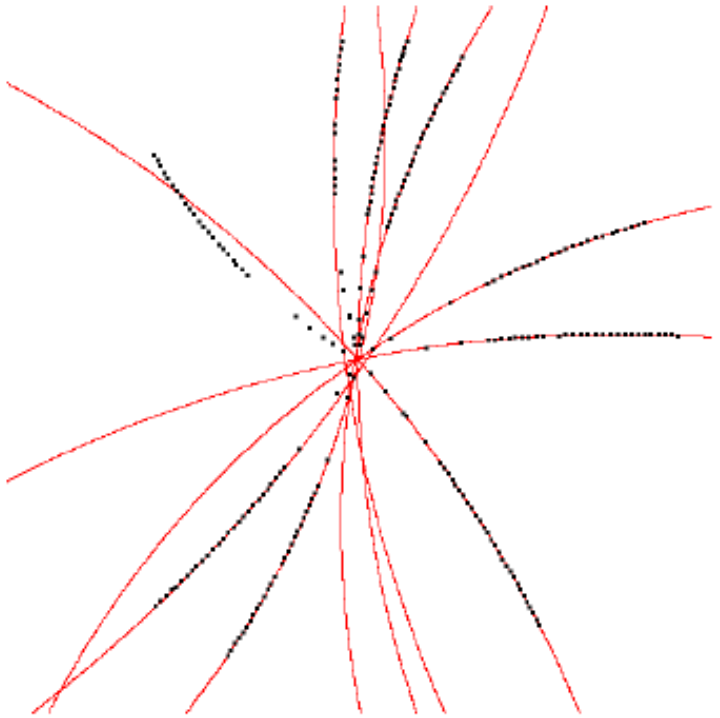


Algorithm: Helix Track Finder + Fitter for Panda



David Münchow

results from emulation



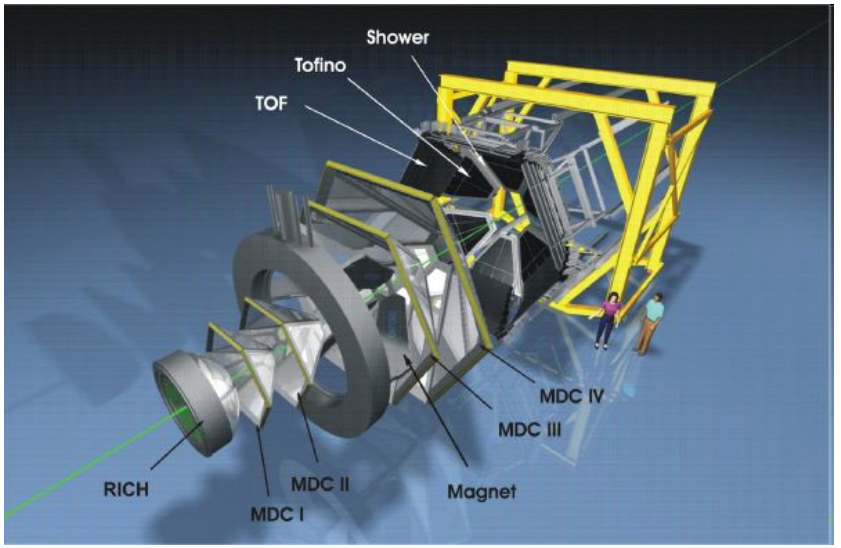
implementation to FPGA = ongoing work

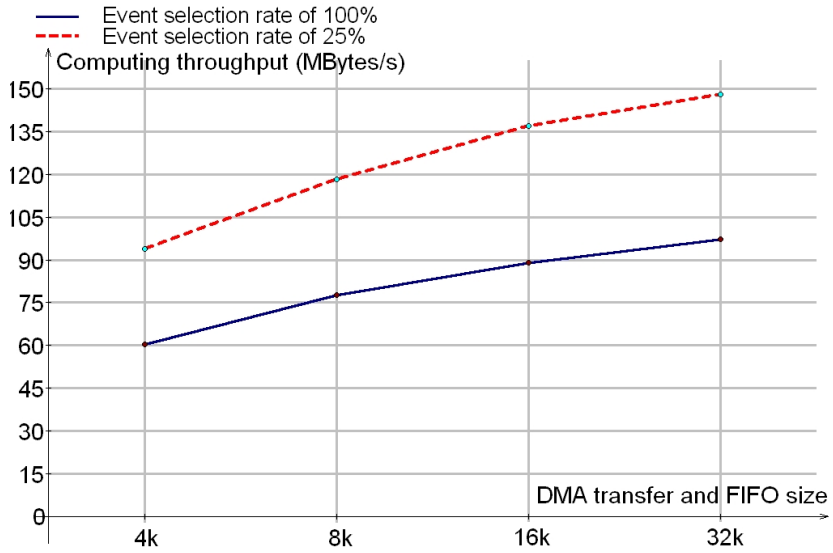
(but lookup tables already used in emulation)

- fix point instead of float
- 24 bit (in division and multiplication 48 bit)
- Hough space of 512×512 indices
- lookup table for sinus(): 128 indices x 16 bit

Additional Algorithm Development all incl. FPGA implementation (work finished)

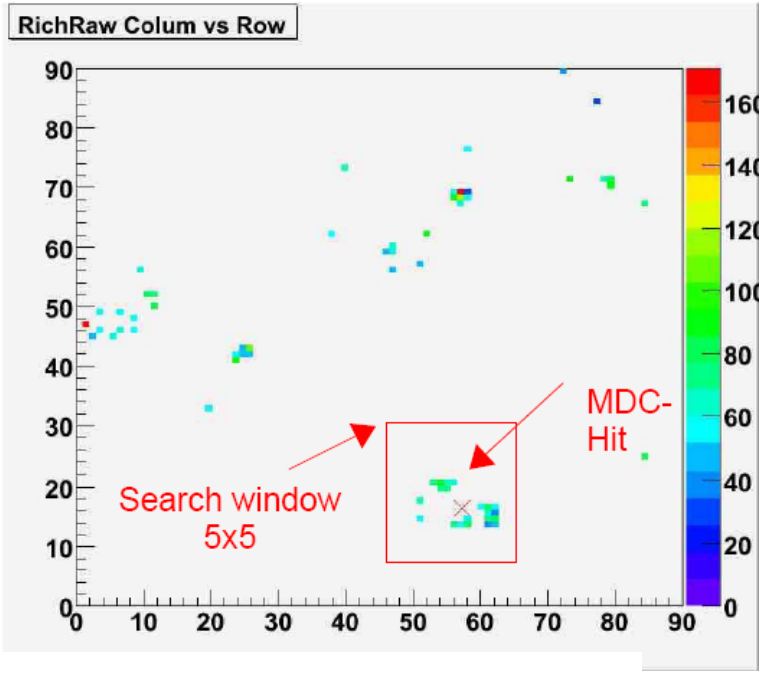
- **Ming Liu - HADES**
  - drift chamber track finder (straight line, <12 wires per track)
  - compare speed to C program running on Xeon 2.4 GHz as software reference
  - different wire multiplicities (10, 30, 50, 200, 400 fired wires out of 2110)
  - speedup of 10.8 – 24.3 (compared to reference)
  - Implement 2-3 cores in parallel on FPGA
- **Shuo Yang – HADES**
  - read HADES binary events from DDR2
  - decode
  - issue accept/reject
  - discard or write back to DDR2
  - achieved ≃ 100 MB/s for 32 kB fifo and DMA size (for 100% event accept)

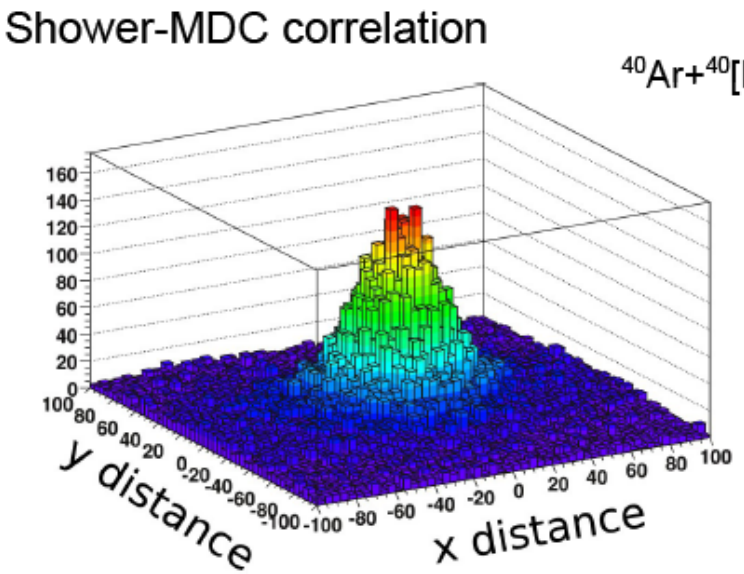




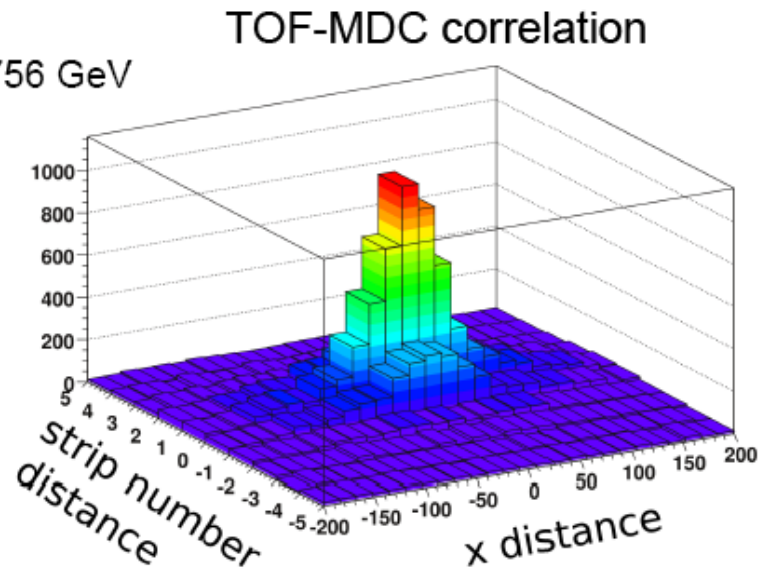
Additional Algorithm Development all incl. FPGA implementation (work ongoing)

- **Johannes Roskoss – HADES**
  - RICH ring finder
  - match ring to drift chamber track (straight line, <12 wires per track)
- **Andreas Kopp – HADES**
  - drift chamber track incl. momentum kick in dipole field
  - match to TOF (2-sided read out scintillator paddles)
  - match to EM Shower detector

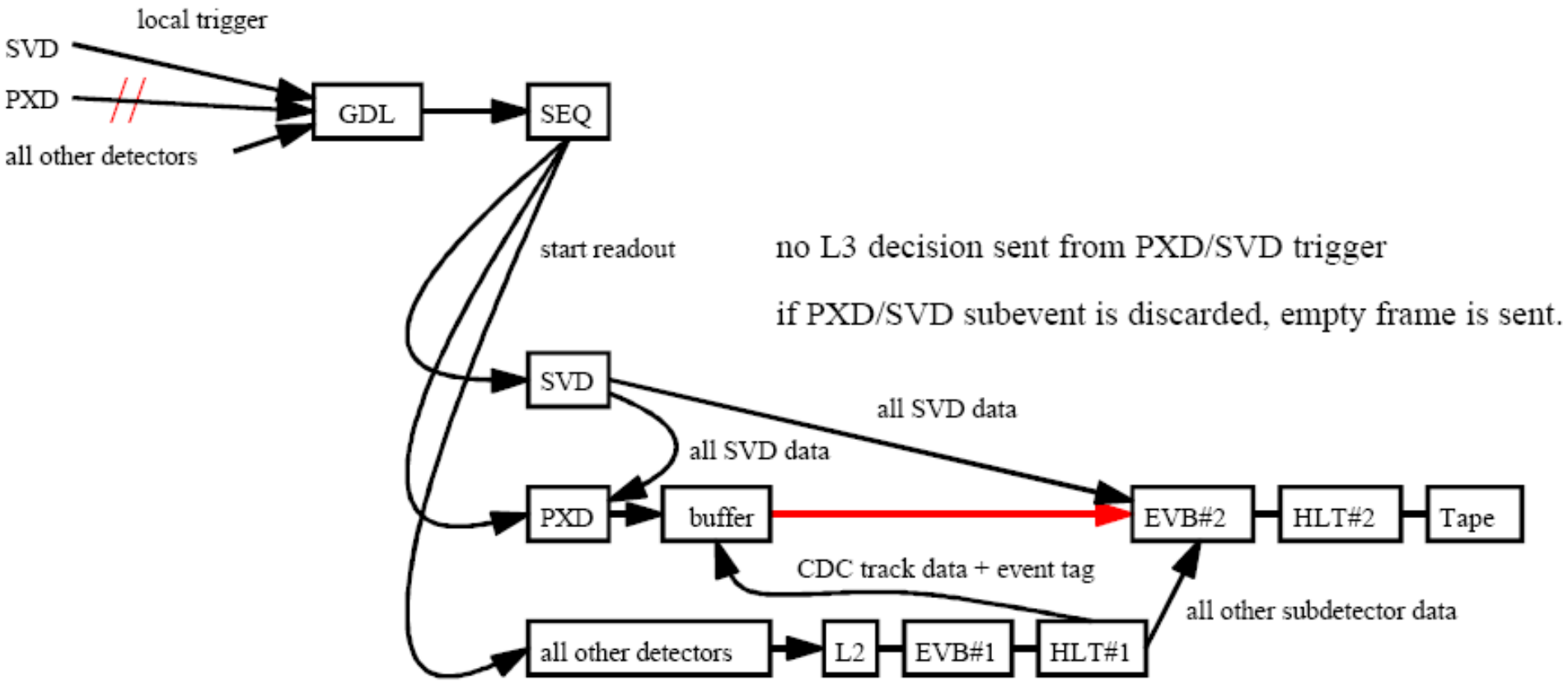




^{40}Ar+^{40}[KCl] @1.756 GeV



PXD Trigger Scheme, 2nd Idea



PXD buffer also needs event sorting.

= reduced PXD data

(by factor 10-100)

HLT = High Level Trigger

HLT#1 = maybe L3

HLT#2 = maybe L4

HLT#1 runs CDC track finder

PXD runs PXD+SVD track finder

Plans for the Near Future

- **Testing Compute Node Version 2.0**
  - **testing FPGA-FPGA communication**
  - **testing board-board communication via backplane**
  - **testing high speed optical link $\leq$ 5 Gbit/s**
- **A major important test and milestone: test sending data from pixel DHH to compute node (similar to our tests with the HADES TDC board) $\rightarrow$ maybe until end of this year?**
- **Algorithm:**
  - **quantitative track finder timing incl. data moving, algorithm, parallelisation and pipelined**
  - **Test with MC data (MPI)**
  - **Test with real SVD 2.0 data**