DAQ Plans for Hyper-Kamiokande

Giles Barr[1], Francesca Di Lodovico[2], Helen O'Keeffe[3]

1 = University of Oxford, 2 = Queen Mary University of London, 3 = Lancaster University

Hyper Kamiokande Open Meeting

January 2014

Outline

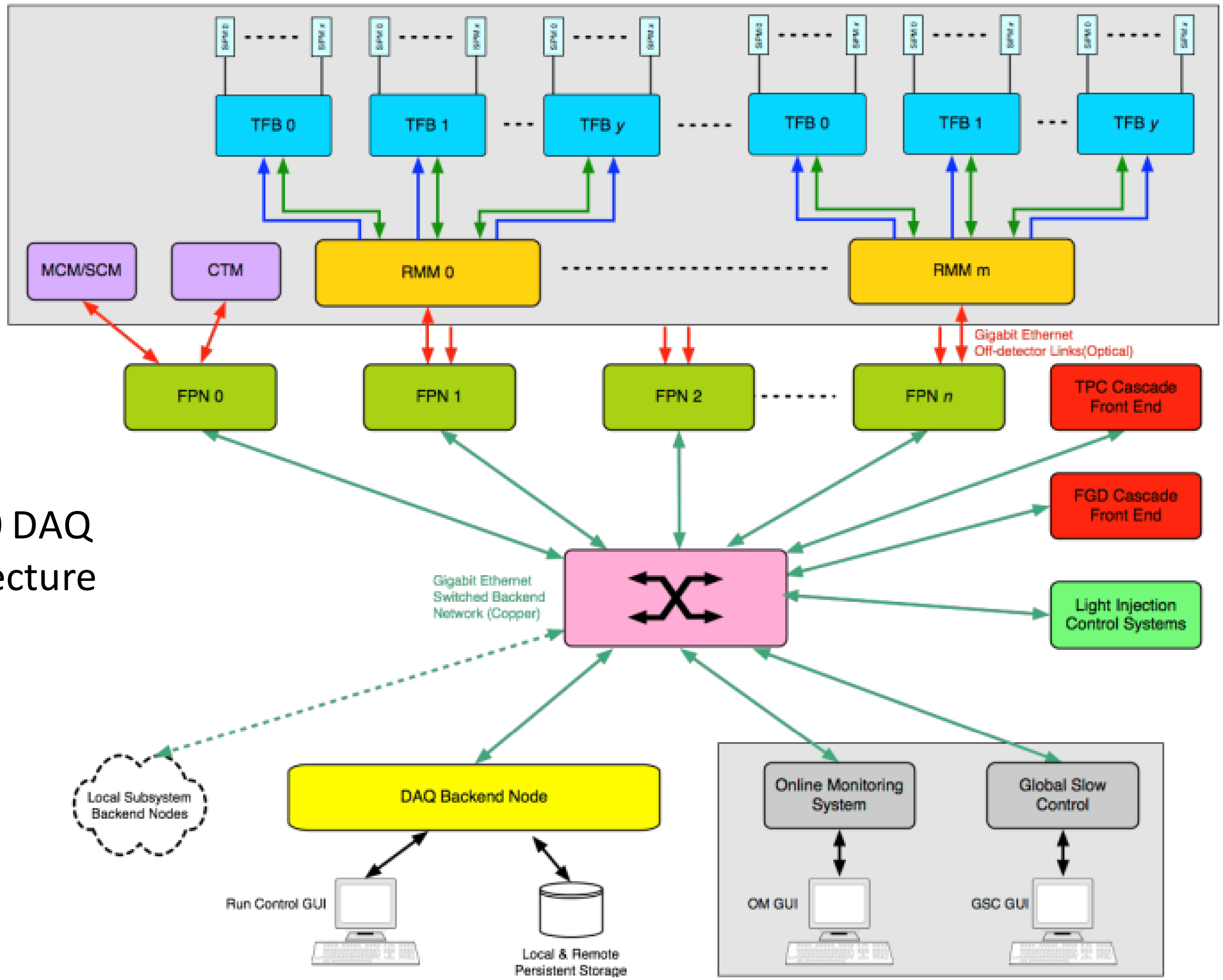
- Contributions to T2K
- Interested institutions
- Interests in HK DAQ

T2K DAQ systems

- UK institutions designed majority of near detector DAQ systems
- Used MIDAS framework
- Incorporated several pieces of custom software for data readout
- Readout of voltage, temperature of electronics boards and interface with global slow control
- Wrote online monitoring software

Continue to play a major role in on-site support, operations, maintenance etc.

T2K DAQ architecture



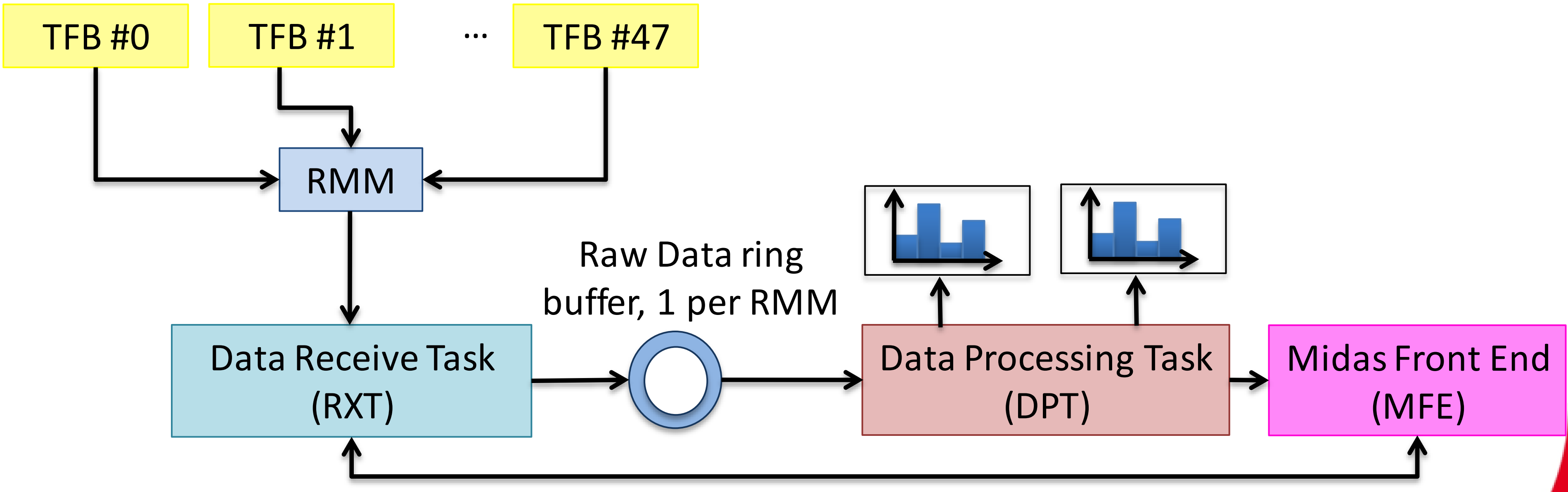
ND280 electronic architecture

fpn = front-end processing node

ND280 DAQ architecture

FPN processes

- Data from electronics boards is retrieved using custom programs written by experts at STFC Rutherford Appleton Laboratory/Oxford



Interested institutions

T2K DAQ group was Oxford and Rutherford, for HK several new institutions will participate:

- **Lancaster University**
- **Queen Mary, University of London**
- STFC Rutherford Appleton Laboratory
- University of Oxford
- **University of Warwick**

Small amount of bridging funds to spend between January 2014 and September 2014

UK interests

Interests are in the following areas

- Design and development of DAQ for Hyper-Kamiokande (and Intermediate Detector). In this talk studies mainly focussing on Hyper-K.
- Readout of electronics and design of backend systems
- Triggering
- Participation in 1 kT prototype
- Online monitoring of data

Work has so far focused on

- Assessing data rates
- Triggering

Data rates

Important to consider rate in full detector volume

- Background rates based on SK levels
- 10 kHz dark noise per PMT
- Assume 12 bytes per PMT hit, 100,000 PMTs in whole detector, 10 hits per background event

Event class	(estimated) rate (Hz)	Estimated data rate
PMT noise	10×10^3 (per PMT)	12 GB/s
^{238}U chain	158	20 MB/s
^{232}Th chain	475	57 MB/s
^{222}Rn	2772	332 MB/s

External events have been ignored

Dark noise may be more optimistic

Accidentals

Expression from SNO technical report SNO-STR-90-036

$$A = \frac{\tau^{-1} k^{n_t} e^{-k}}{(n_t - 1)!}$$

where

n_t = number of tubes firing

N = total number of PMTs $\rightarrow$ 10,000 per compartment

R = Dark noise rate of the tubes = 10 kHz

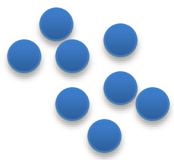
τ = discriminator time width = 100 ns

$k = NR\tau$

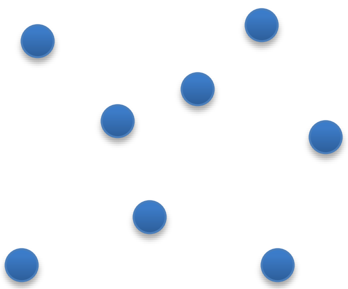
Nhit threshold	Accidental rate
10	12.5 MHz
15	5.2 MHz
20	373 kHz
25	7.3 kHz

Reducing accidentals

- Dark noise affects where the data-taking threshold is
- Are there ways to reduce the effect of dark noise, allowing us to access more low energy physics for a small cost?
- Could use following
  - Timing - dark noise hits are random within a trigger window
  - Charge
  - Angular distributions – dark noise more isotropic in detector (experience from SNO)



Low E event
Average angle between hits is small



Noise “event”
Average angle between hits is larger

Event readout + triggering

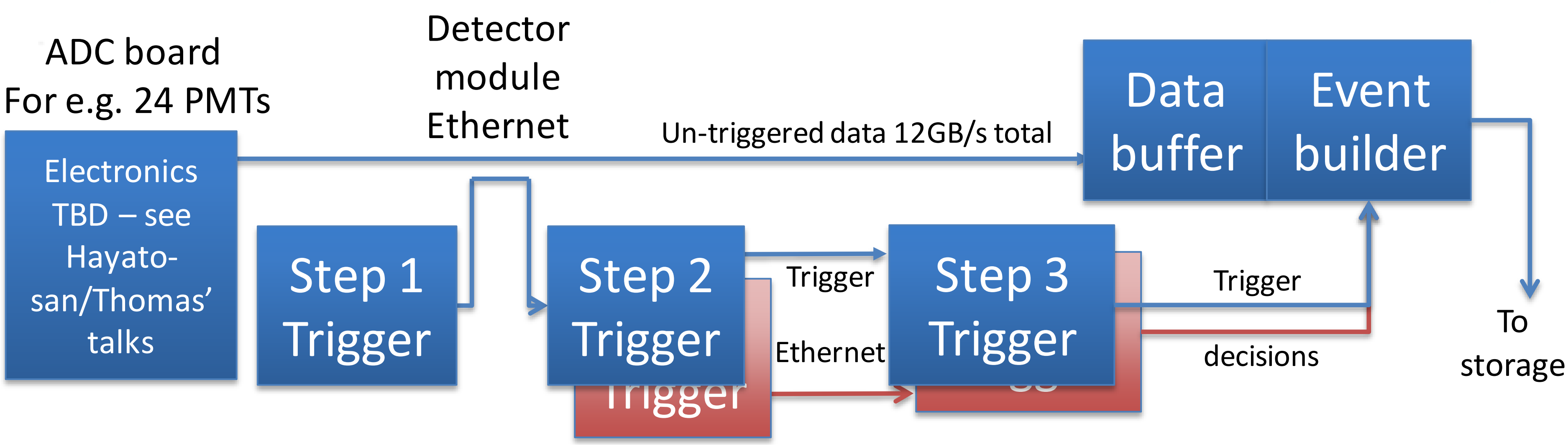
- Main trigger strategy – like Super-K: Nhits
  - Good enough for atmospheric events, proton decay, beam neutrinos
- Dominated by dark noise coincidences
  - Limits threshold for low energy physics
- Additional "smarter" trigger to separate low energy neutrino events from random dark noise coincidences
  - Important to attempt to reduce solar neutrino threshold
  - Helps with supernova physics

Nhits trigger overview

Step 1	• **Calculate quantities in local pad** • e.g. 6x4=24 PMT or 8x8=64 PMT (depending on readout board) • **Linear (additive) variables, simple to compute**	• Close coupled to readout, handshake ≈ 1usec • Ideal for FPGA design
Step 2	• **Aggregate local pads into pads** • **“Displaced pads” concept** • Duplicate steps 2 and 3 to avoid spatial boundary inefficiencies	• Transfer step 1 to step 2 over Ethernet. Canadian redundant path idea. • Algorithm easy if step 1 quantities are additive

Step 3	• **Now have ≈ 50-200 pads in each compartment of HK**
	• **Physics motivated cuts:**
	    • <u>Always accept if Nhits > N</u>
	    • <u>Accept more events with low energy signatures</u>
	        • Local coincidence
	        • Trilateration reconstruction, localize vertex and reduce time coincidence window
		• Step 2 to step 3 over separate Ethernet network,
		• Parallel processing farm
		• Designated node for each 10usec window in time (+0.5usec for overlap)
		• Trilateration may be suitable for SIMD or GPU

Event readout + Triggering

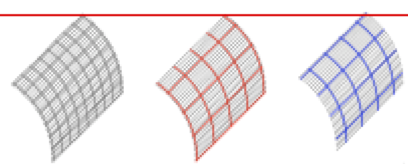


Step 1: On Board Coincidence (OBC)
Trigger processor attached to each readout board

Step 2: Regional Processing Node (RPN)
Rearrange trigger info into 'pads'

Step 3: Trigger Algorithm Node (TAN)
Decisions in farm: Each node sees all data for a given time window

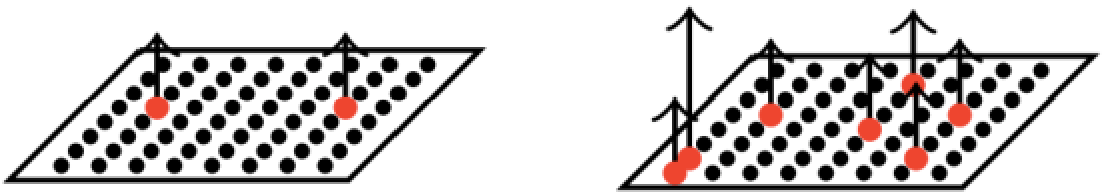
Un-triggered data buffered to wait for trigger decision to build events



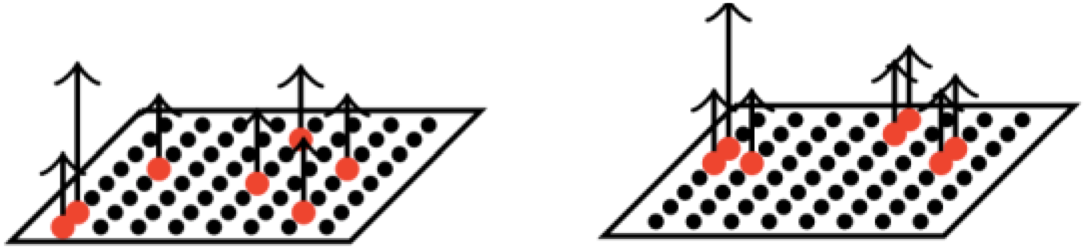
Data buffer could be extended to store Super Nova Cached data: 1 hour = 400TB

Step 1: On Board Coincidence (OBC)

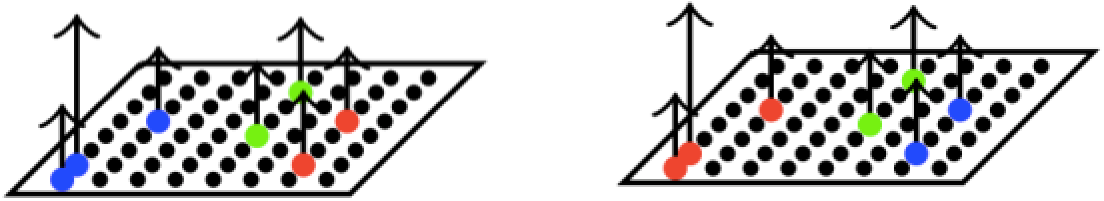
Nhits



Spatial moments



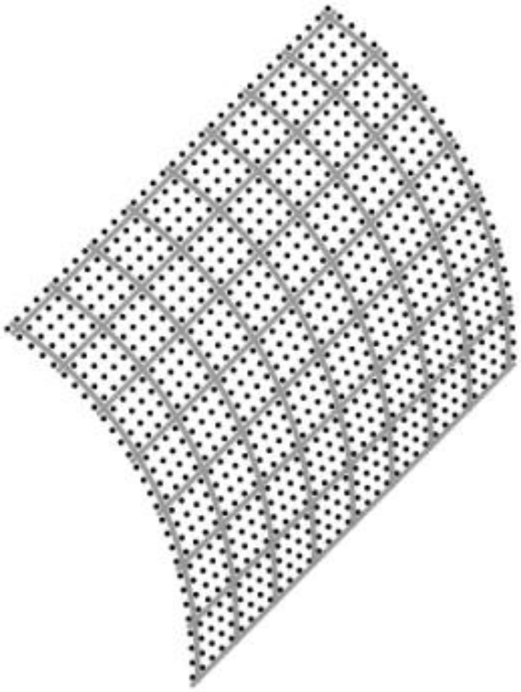
Timing



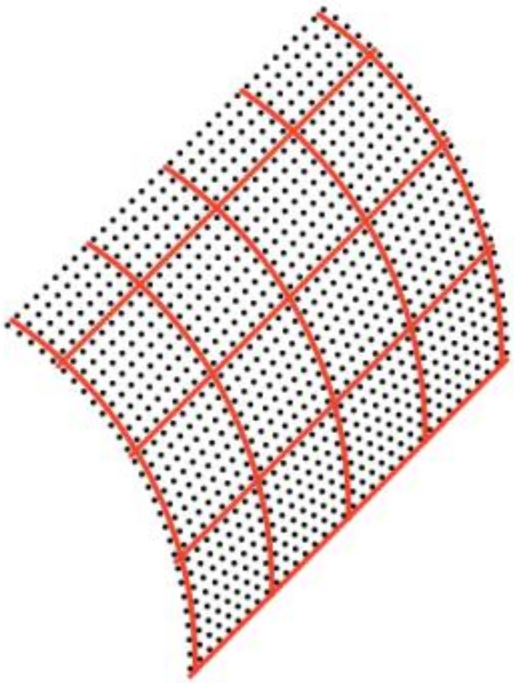
Compute for each local pad, for time bins of O(100ns).

Send local-pad#, overall time, overall PH, Nhits, x and y spatial moment, x and y time moment to trigger

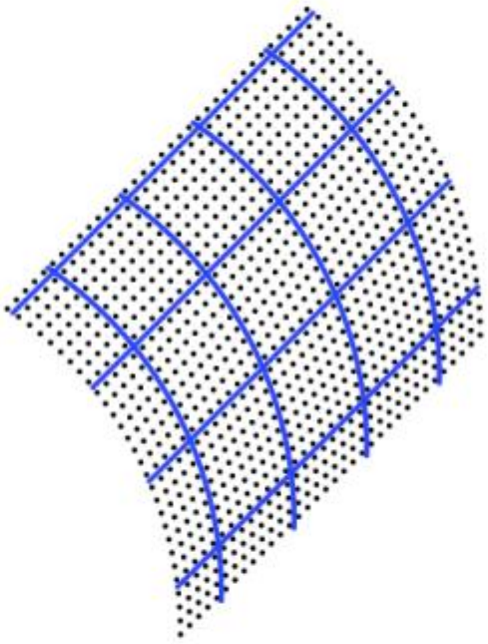
Step 2: Regional Processing Node (RPN)



Local Pads from step 1

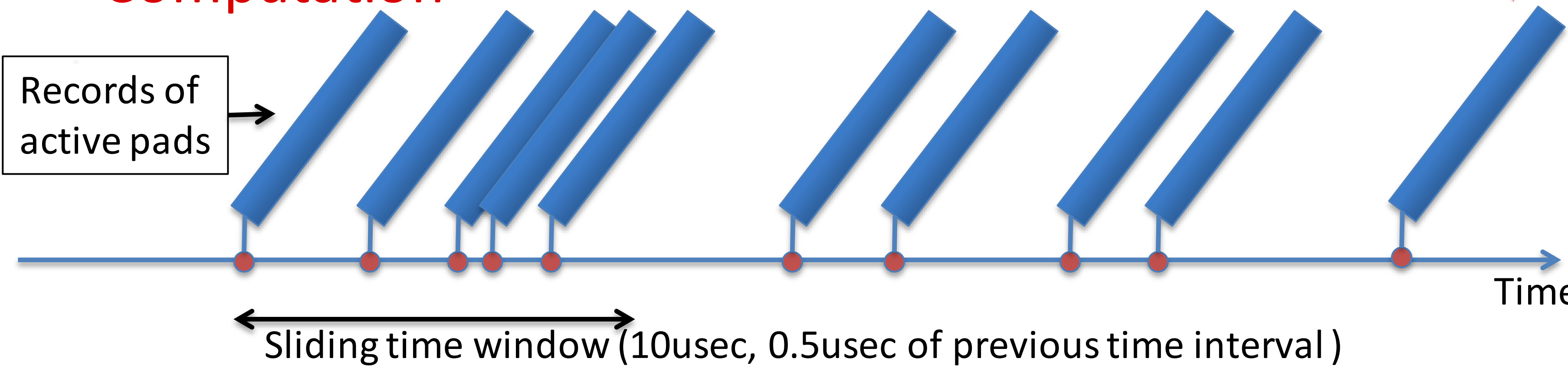


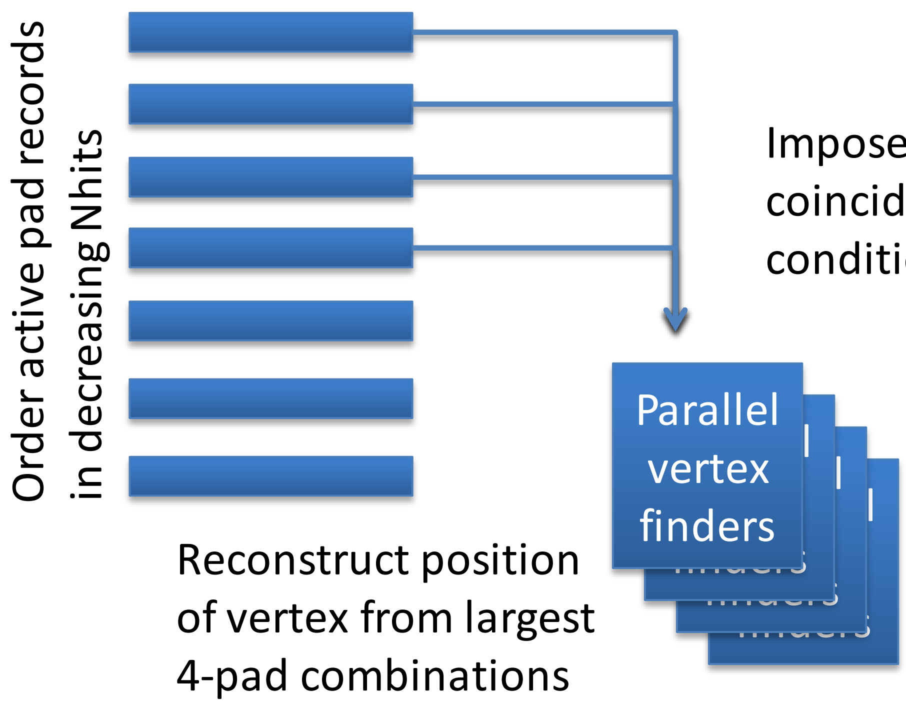
Pads for step 3



Displaced pads for step 3

Step 3: Trigger Algorithm Node (TAN) Computation





Decision 1

If Nhits > N: PASS

Decision 2

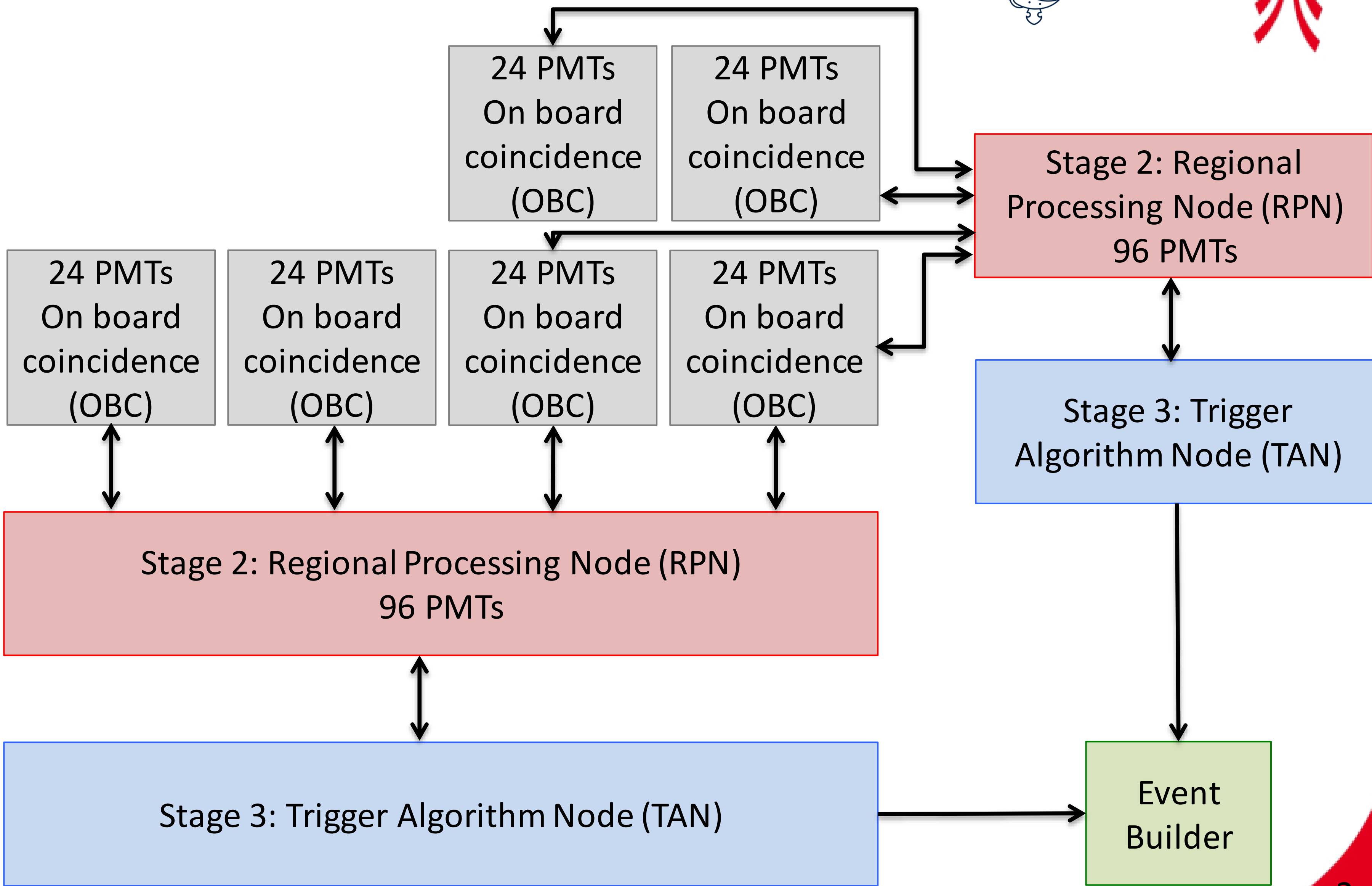
SAVE events with lower Nhits if sufficient hits in tight timing window defined from vertex

Step 3: Trigger Algorithm Node (TAN)

- Processor assigned for each 10usec time interval.
  - Include data from last 0.5usec of previous time interval to avoid problems with time boundaries
- **Events passing Nhits criteria always pass**
- **Collect additional low-energy events with cuts to select clusters and not random noise combinations**
- Sliding window in time:
  - Impose local coincidence requirement
  - Sort pads by NHITS starting with highest
  - Use parallel processing to consider best pad combinatorics
    - Reconstruct vertex with 4 big pads, consider if other pads are consistent

Remember: We don’t need to consider the most complicated combinations - they are accepted by the simple Nhits trigger

Event readout + Triggering



Event readout + Triggering

Stage 1: On Board Coincidence (OBC)
Collect hits from 24 channels

Nhits	**Time**	**ϑ-X**	**ϑ-Y**	**X**	**Y**		

Approximately 420 of these On Board Coincidence, OBCs (number determined by channels read out on electronics board)

Stage 2: Regional Processing Node (RPN)
Combine information from 4 x 24 channels (OBCs)

Nhits	**Time**	**ϑ-X**	**ϑ-Y**	**X**	**Y**		

Combine data from 4 OBCs. Approximately 105 RPNs + 105 RPNs from shifted configuration. 210 RPNs in total

Stage 3: Trigger Algorithm Node (TAN)
Combine information from RPNs and divide data into time slices.

Data from RPN sent to TAN. Approximately 210 nodes required.

Event readout + Triggering

Work package 2: On Board Coincidence (OBC)
Local threshold requirement, implement on xylinx type chip.

Work Package 3: Regional Processing Node (RPN)
Basic algorithm to decide if something interesting has happened.

Work Package 1: Event readout and backend systems

Work Package 4: Trigger Algorithm Node (TAN)
More complex algorithms to regain low energy events.

Work Package 5: Event Builder and communication with offline storage. Data monitoring.

Conclusions

- Significant fraction of HK UK institutes are interested in developing DAQ systems.
- Small amount of money to spend on technical support between Jan. and Sept. 2014.
- Focus on design of data readout systems and triggering
- We can adapt the current schema to deal with more detailed information when known (eg occupancy - the number of PMT hits in a board - for low energy events vs dark rate)
- Participation in 1kT prototype – useful to test algorithms

Backup slides