Network Processors in the LHCb DAQ System

Presentation given at the
Atlas Trigger/DAQ Week's
Readout Sub-System Session
July 2001

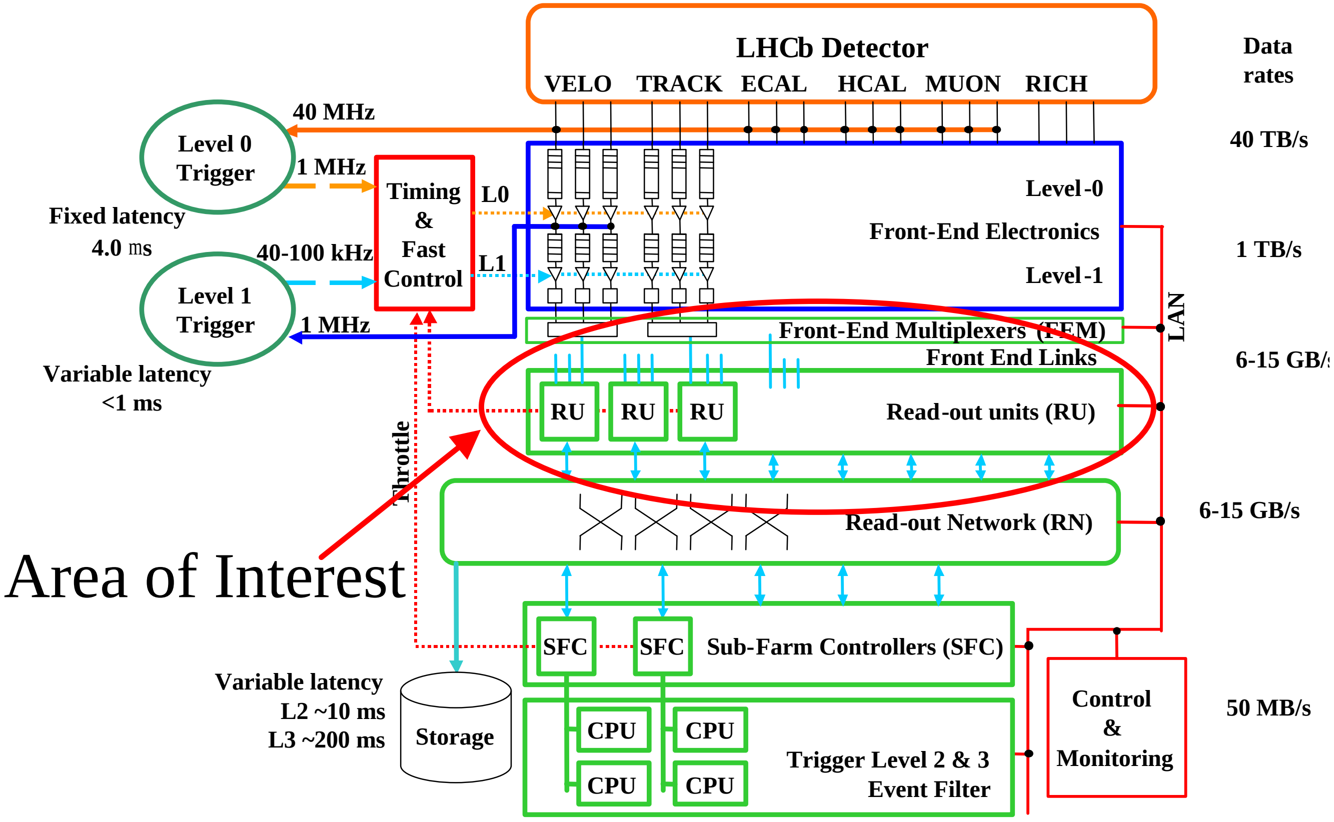
Beat Jost & Niko Neufeld
Cern / EP

Outline

- ❑ LHCb Introduction
- ❑ Network Processor Introduction
- ❑ Application to Data Multiplexing/Merging
- ❑ Performance for Data Multiplexing/Event-Building
- ❑ Board-Level Integration – first ideas
- ❑ Plans
- ❑ Conclusion

LHCb Introduction - Architecture

LHCb DAQ Architecture



Data-Flow Requirements:

•L1-Rate: 40-100 kHz

•Event Size: ~150 kB

•Data Rate: 6-15 GB/s

Multiplexing factors

In FEM/RU: ~4-~16:1

LHCb Introduction - Protocol

The protocol for the data flow of LHCb is a pure push-through protocol, i.e. every source of data sends them on as soon as available.

Motivation:

- ➢ Major simplification of the individual components (important for large numbers)

Complete events are transferred to CPU farm for software triggering

Motivation:

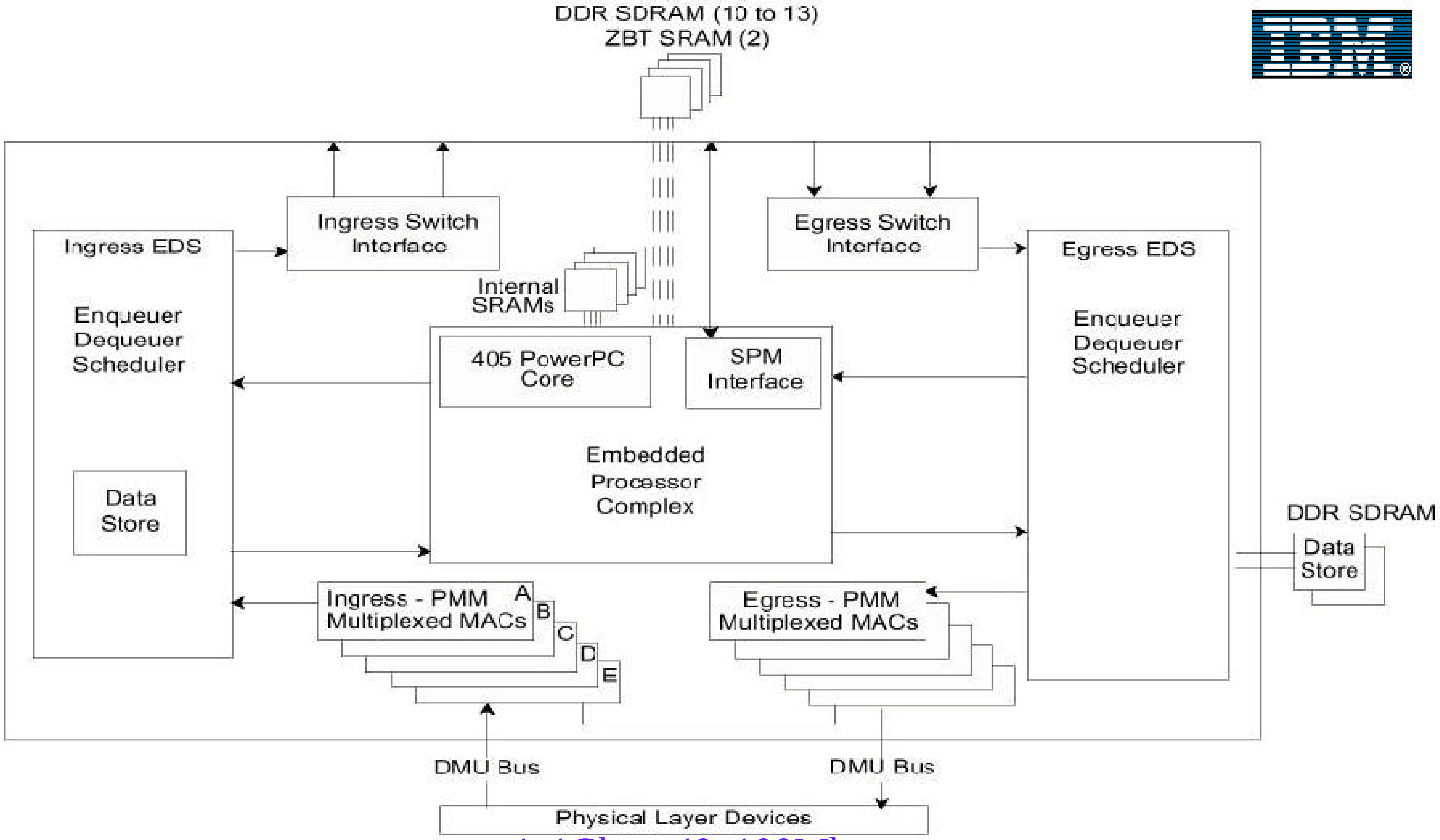
- ➢ Full flexibility and efficiency of the software triggers, without prejudice on the readout protocol, at the cost of higher bandwidth needed

Introduction to Network Processors

- ❑ Network Processors are a new technology gaining very much in momentum in the switch industry. All major chip manufacturers are working on them (IBM, Intel, Motorola, …)
- ❑ Target market are switch manufacturers using them as input stage of high-speed switches.
- ❑ Consist of a set of RISC core processors (usually multithreaded in hardware) with specialized co-processors for functions like tree-lookup or checksum calculations, all on one chip
- ❑ RISC processors are specialized at frame manipulations

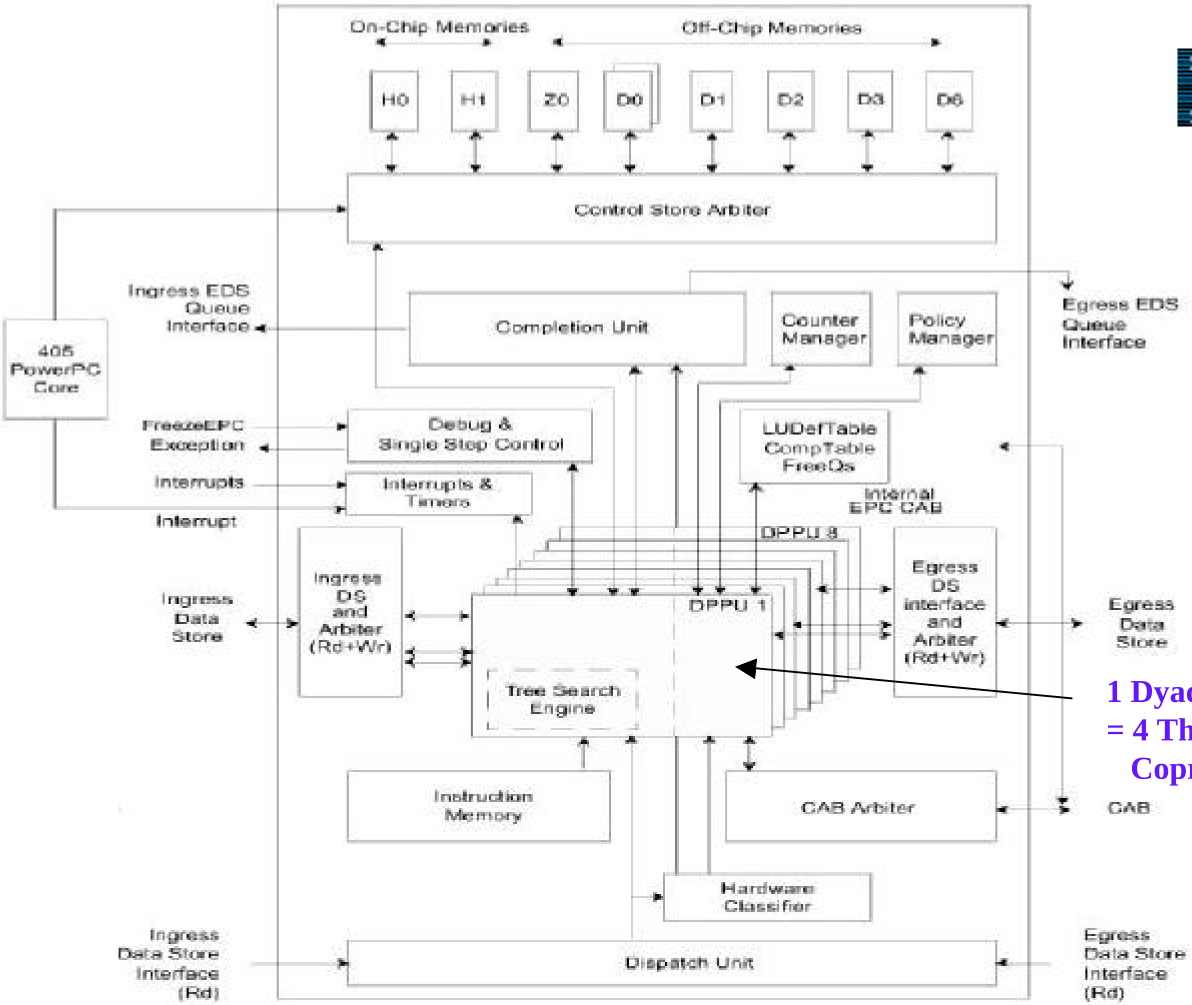
- ❑ We somehow abuse them for doing event-building (assembly of several data frames to one bigger one) in networked DAQ systems
- ❑ We focus for the time being on the IBM NP4GS3(B) Network Processor

NP4GS3 - General Architecture



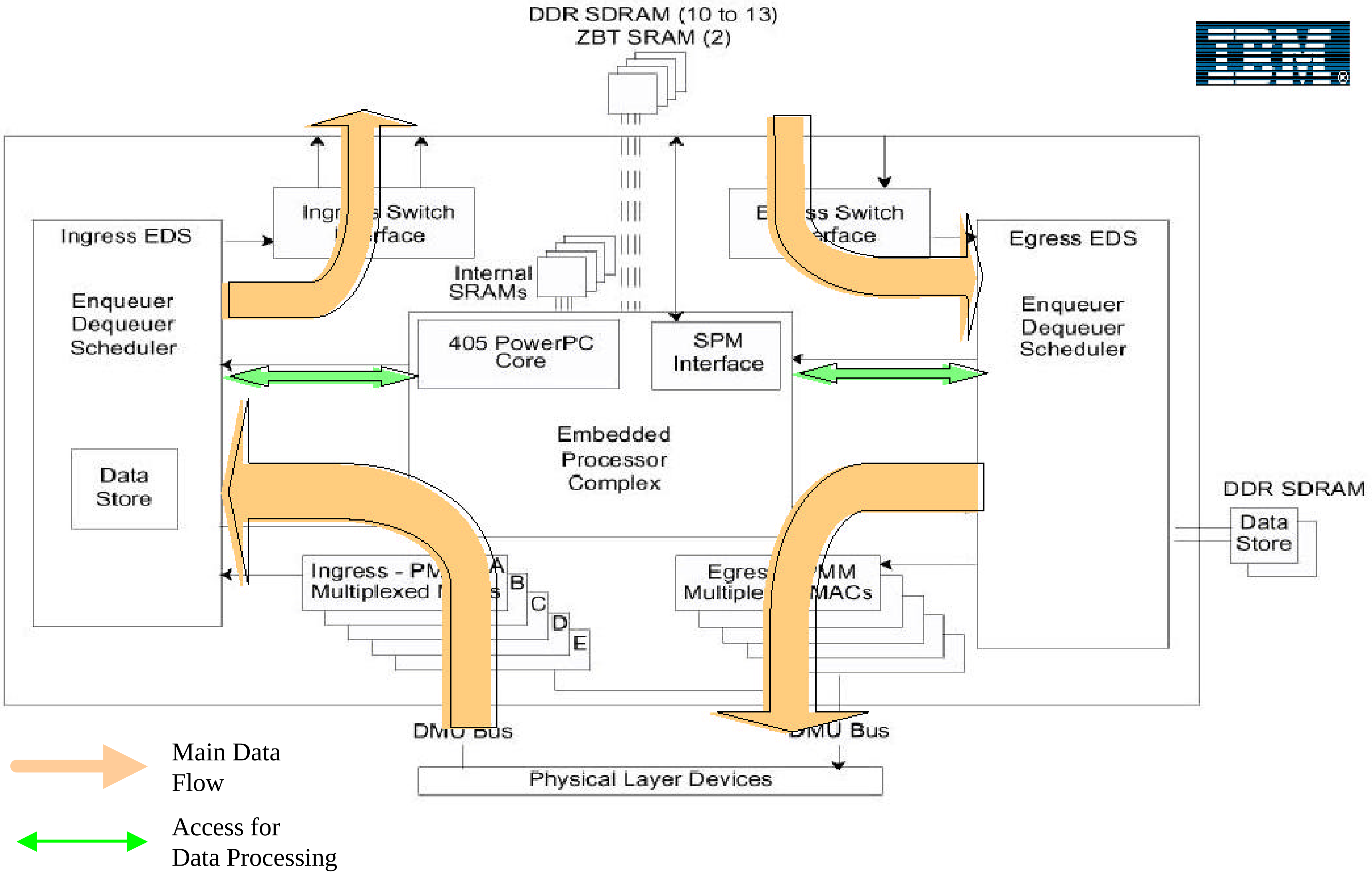
4x1Gb or 40x100Mb

NP4GS3 - Embedded Processor Complex



1 Dyadic Protocol Processor Unit = 4 Threads sharing one set of Coprocessors

NP4GS3 - Data Flow



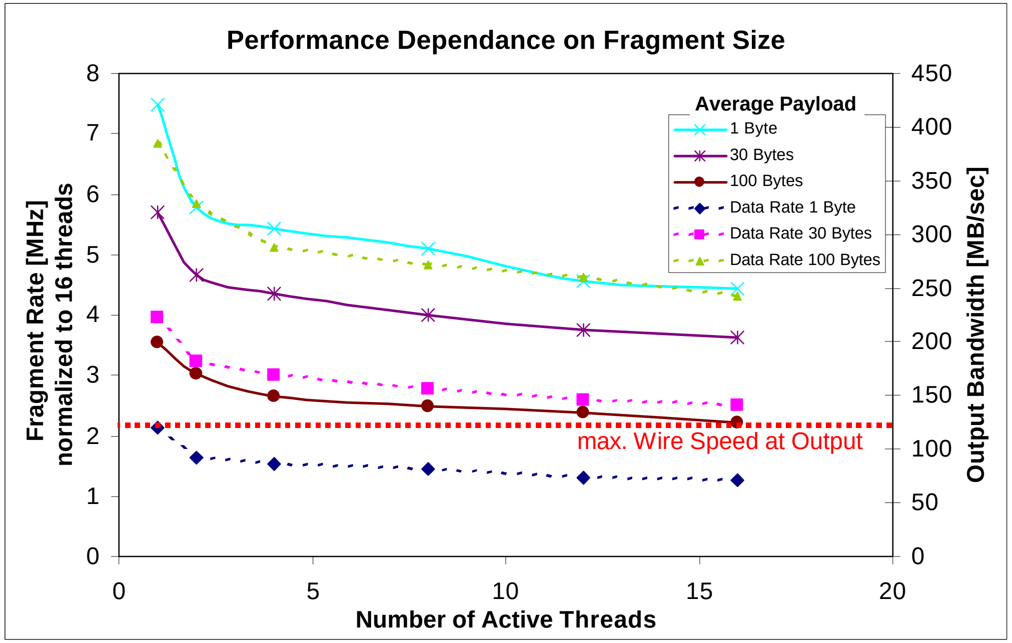
Development Environment and Experience

- There is a very elaborate development environment available, consisting of
  - Assembler
  - Simulator/Debugger
  - Profiler for performance studies
  - Reference hardware kit (equivalent in functionality to what we want to have on a board)
- Our experience is very positive
  - Without the simulator it is impossible to develop and test code (specially if there are problems with synchronization)
  - The performance measurements need to be confirmed with real hardware
  - There are still a few undesired features that will hopefully be ironed out eventually.

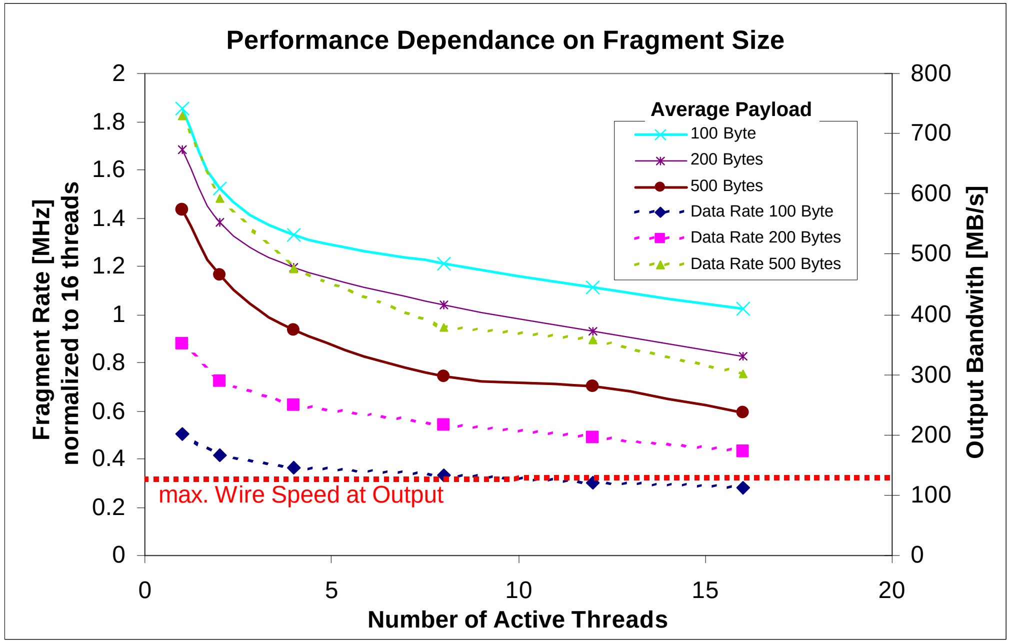
Performance for 4:1 Event-Building

Two versions of the code written, debugged and simulated (cycle precise) taking into account contentions for shared resources

Optimized for very small incoming fragments (30-60 Bytes)

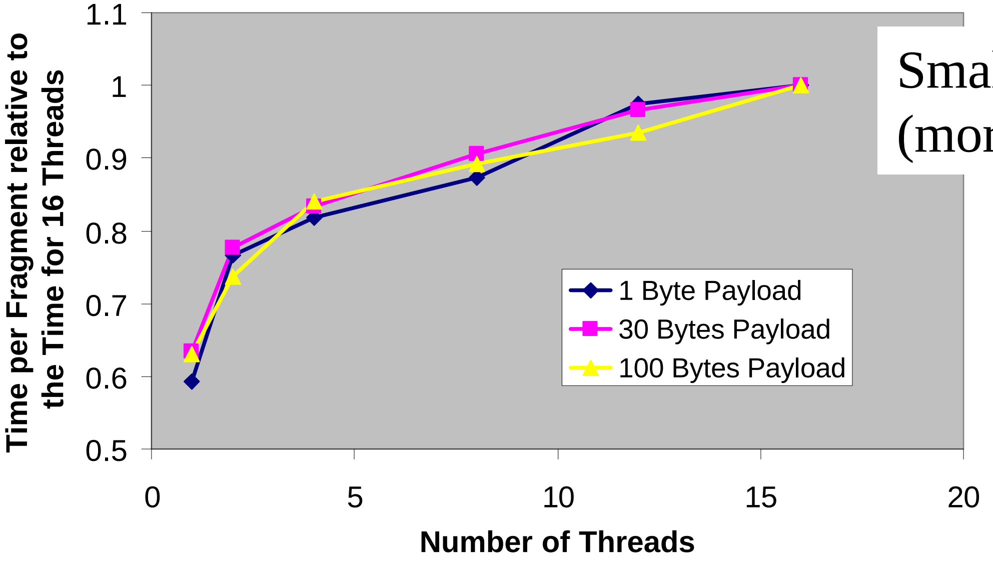


Optimized for larger incoming fragments (~500 Bytes)



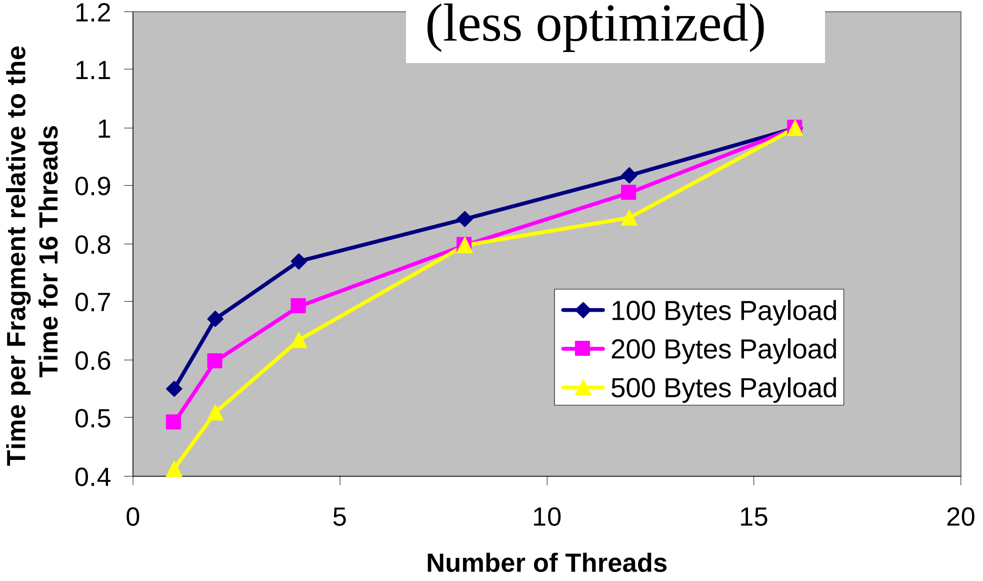
→For all practical purposes we achieve wire-speed event-building performance

Scaling…



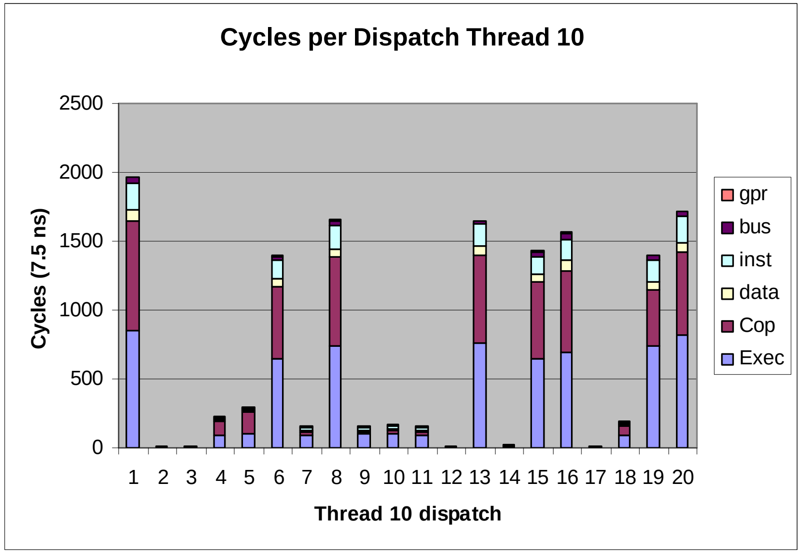
Small Fragments
(more optimized)

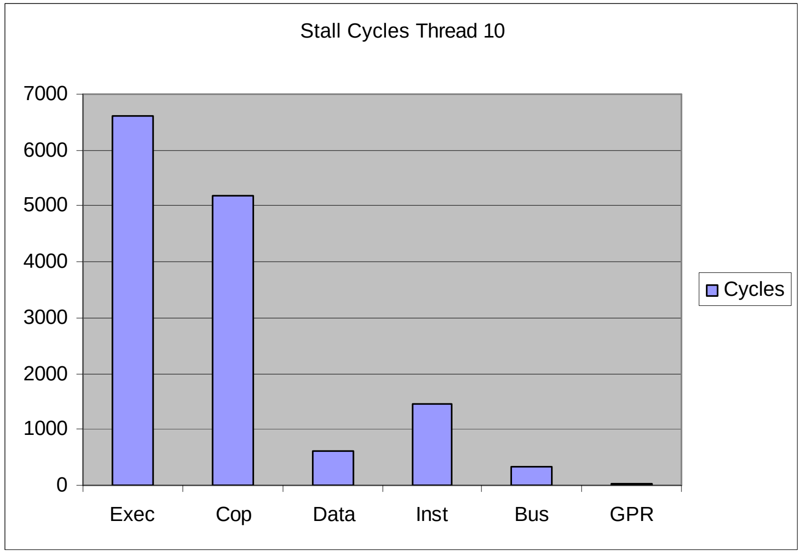
Larger Fragments
(less optimized)

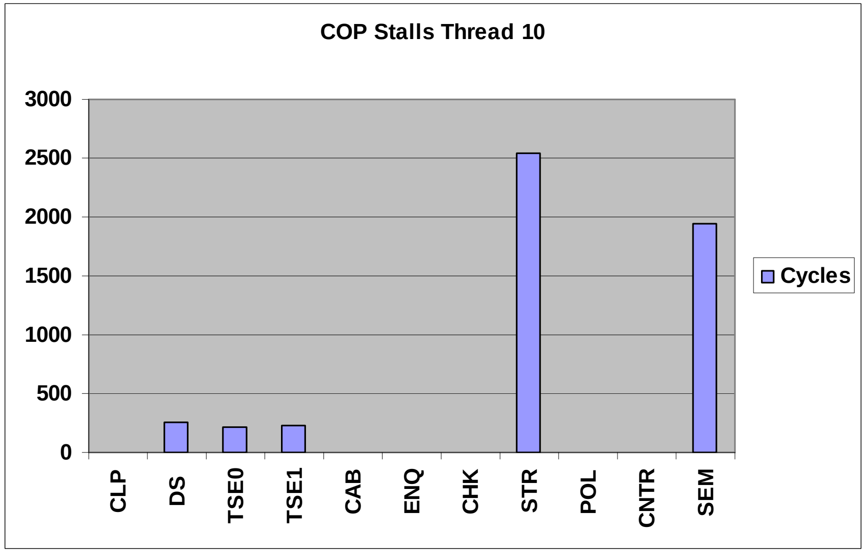


Reason for non-scaling is contention for common resources (e.g. memory buses, coprocessors,…)

Profiler Information

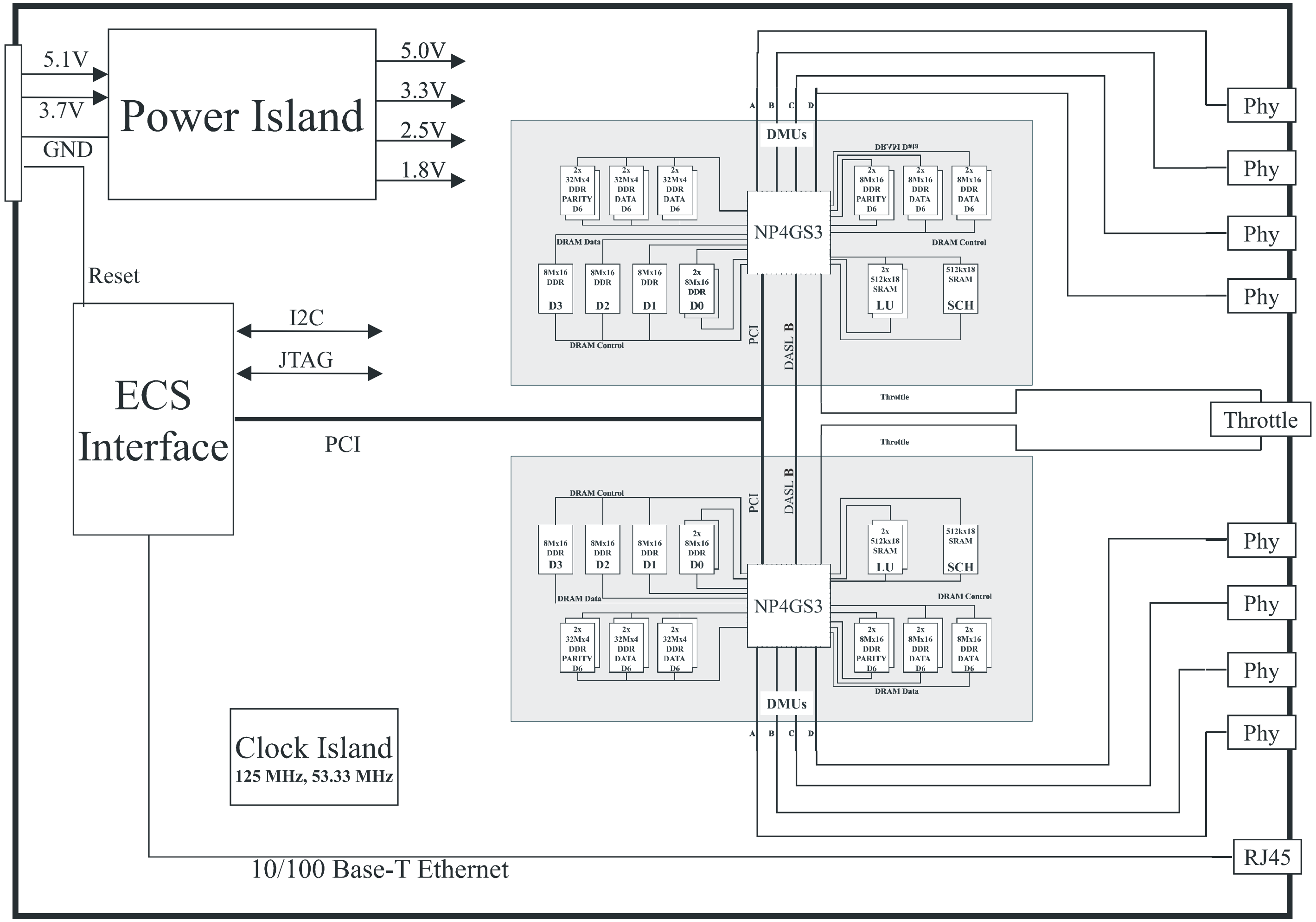




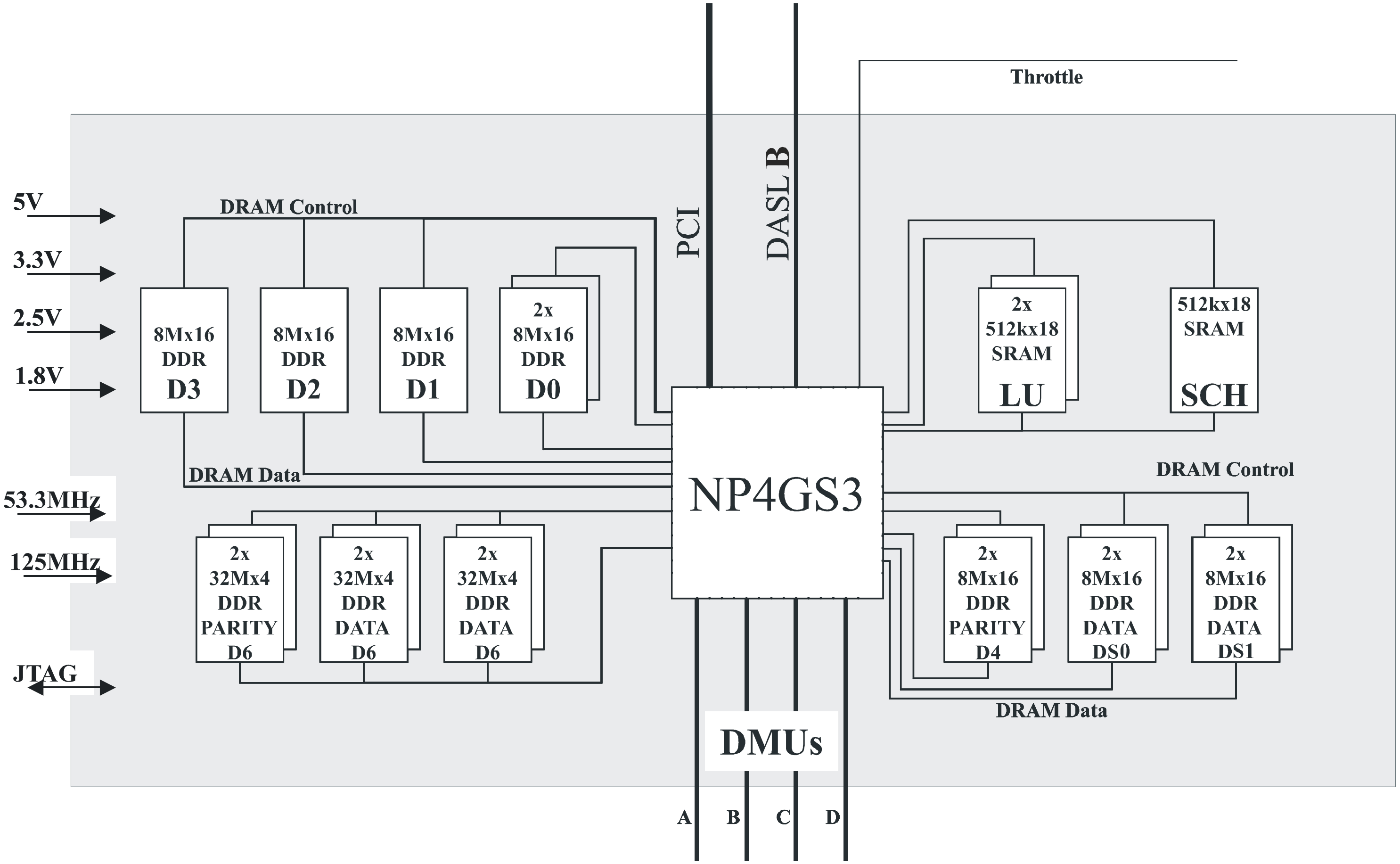


First Ideas on Board-Level Integration

Carrier Board with all the infrastructure (Power, Clock) and the link to the controls system plus mezzanine cards holding the NPs



Mezzanine Card Architecture

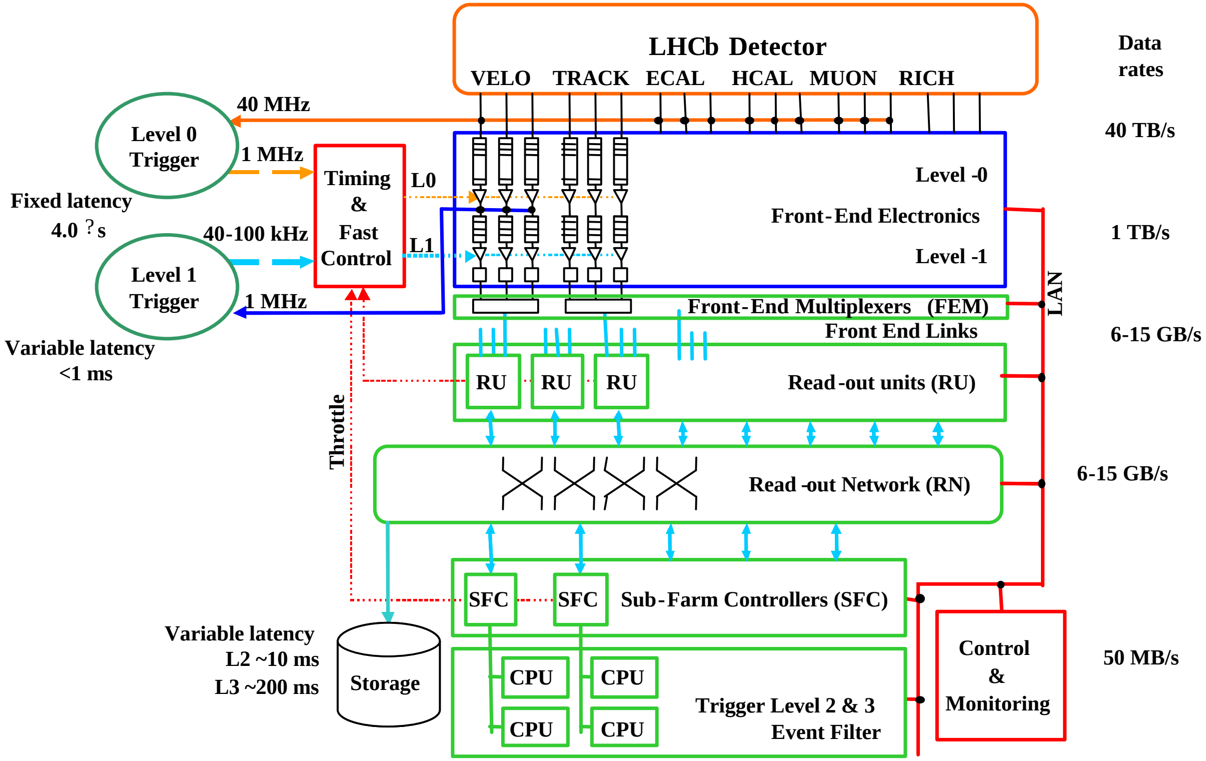


Benefits:

- Most complex parts confined
- Much less I/O pins (~300 compared to >1000

Applications in LHCb (potential)

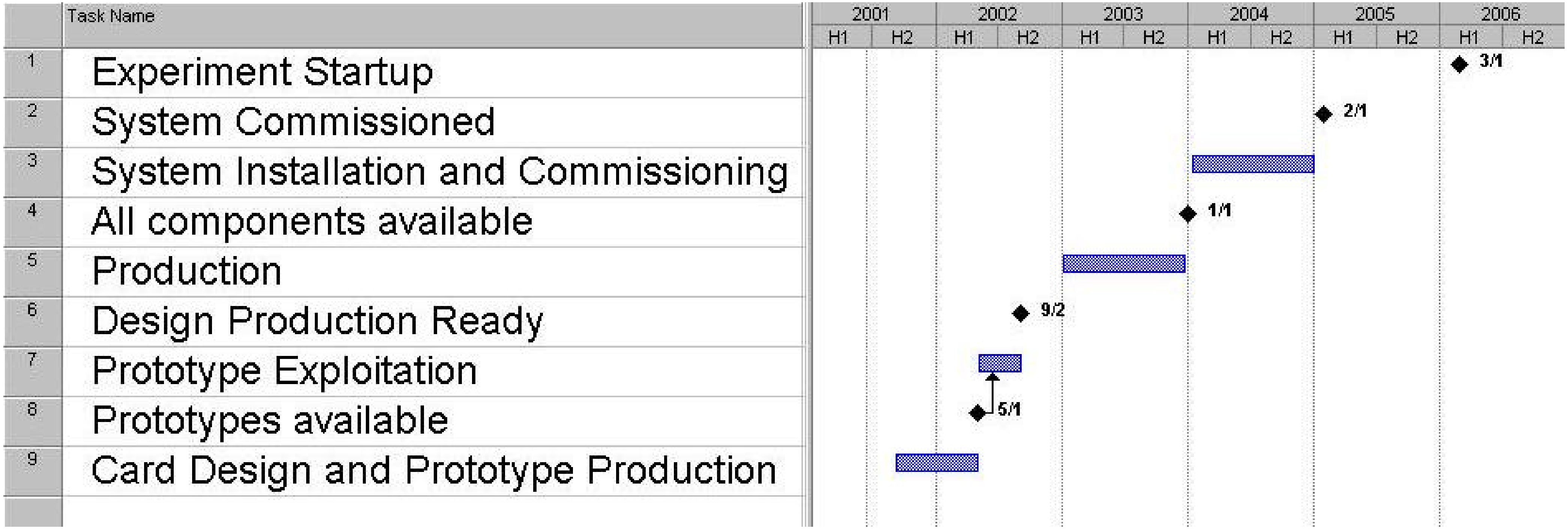
- ❑ The module envisaged is very generic. It could be used for
  - ➢ Front-End Multiplexing/Readout Unit
  - ➢ Building block for the readout network (8-port switch)
  - ➢ Final event-building element downstream of the readout network as a replacement of "smart NICs"
- ❑ Uniform Hardware. The software loaded determines the functionality
- ❑ Of course there is a bias to GbE in the approach -> need Slink implementation over GbE



Plans

- ❑ Acquired a reference kit to verify measurements on real hardware. Remember that the reference kit is functionally equivalent to the hardware outlines before.
- ❑ We are negotiating the design of the mezzanine card with several companies (COST!!!)
- ❑ Internal review of the LHCb FEM/RU complex on July 24
  - ➢ Alternative proposals for RU (FPGA-based, NP-based)
  - ➢ Criteria will be performance, flexibility, maintainability, cost
- ❑ Decision on base-line option before September 2001
- ❑ TDR submission end 2001 with baseline option.

Overall LHCb Planning concerning NPs

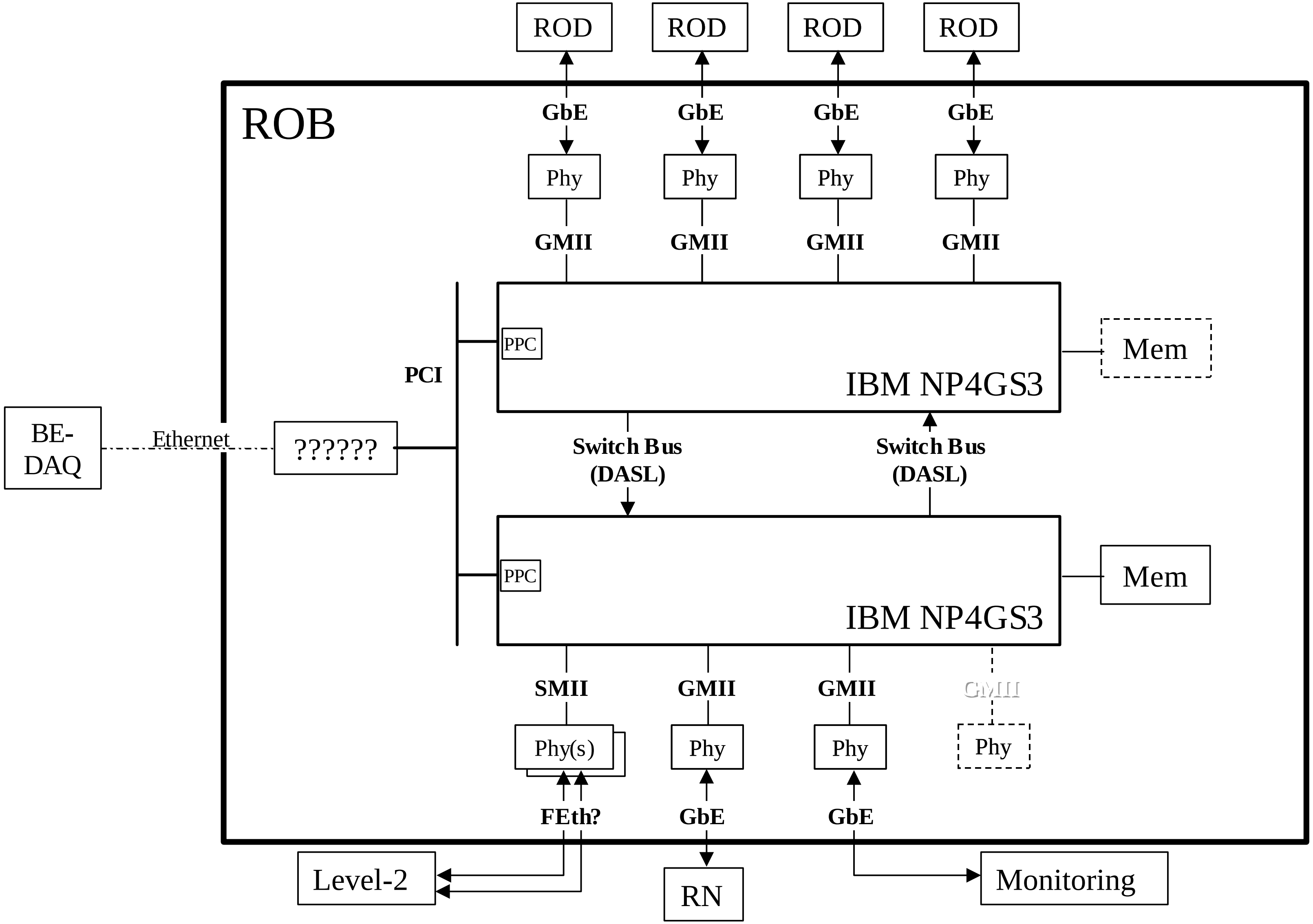


	Task Name
1	Experiment Startup
2	System Commissioned
3	System Installation and Commissioning
4	All components available
5	Production
6	Design Production Ready
7	Prototype Exploitation
8	Prototypes available
9	Card Design and Prototype Production

Conclusion

- ❑ Network Processors are a promising technology to be applied to network-based DAQ systems
- ❑ A very elaborate development environment is available
- ❑ We have outlined a generic module that could serve all functions throughout the LHCb DAQ System
- ❑ The performance achieved with the first version of the code is shown by simulation to be largely sufficient for LHCb and we achieve more than wire-speed performance for all practical purposes
- ❑ Technically NPs are far superior to anything else for the application they are meant for (not completely unbiased…). THE basic 'problem' is to get the cost under control
- ❑ Of course we are ready to collaborate on the use of NPs with everybody at all levels (software, hardware, experience,…)

Possible Application in Atlas



Beat Jost, Cern