Network Processor based RU

Implementation, Applicability, Summary

Readout Unit Review
24 July 2001

Beat Jost, Niko Neufeld
Cern / EP

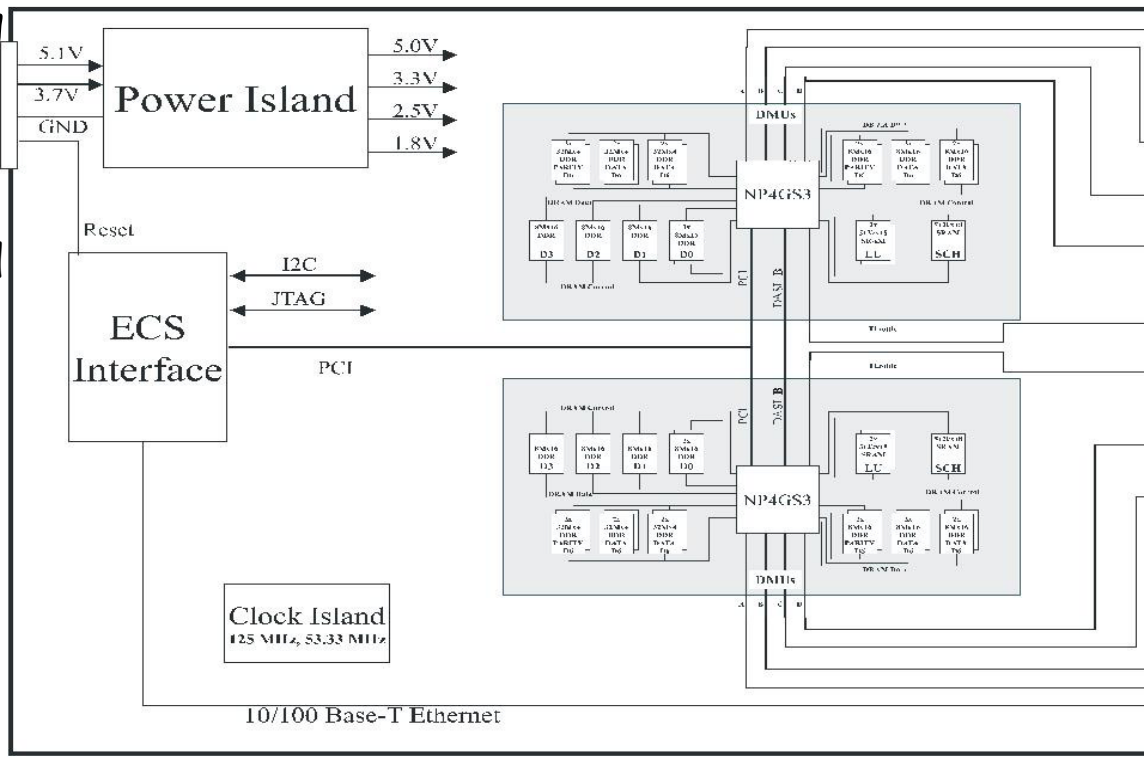
Outline

- Board-Level Integration of NP
- Applicability in LHCb
  - Data-Acquisition
    - Example: Small-scale Lab Setup
  - Level-1 Trigger
- Hardware Design, Production and Cost
- Estimated Scale of the Systems
- Summary of Features of a Software Driven RU
- Summaries
- Conclusions

Board-Level Integration

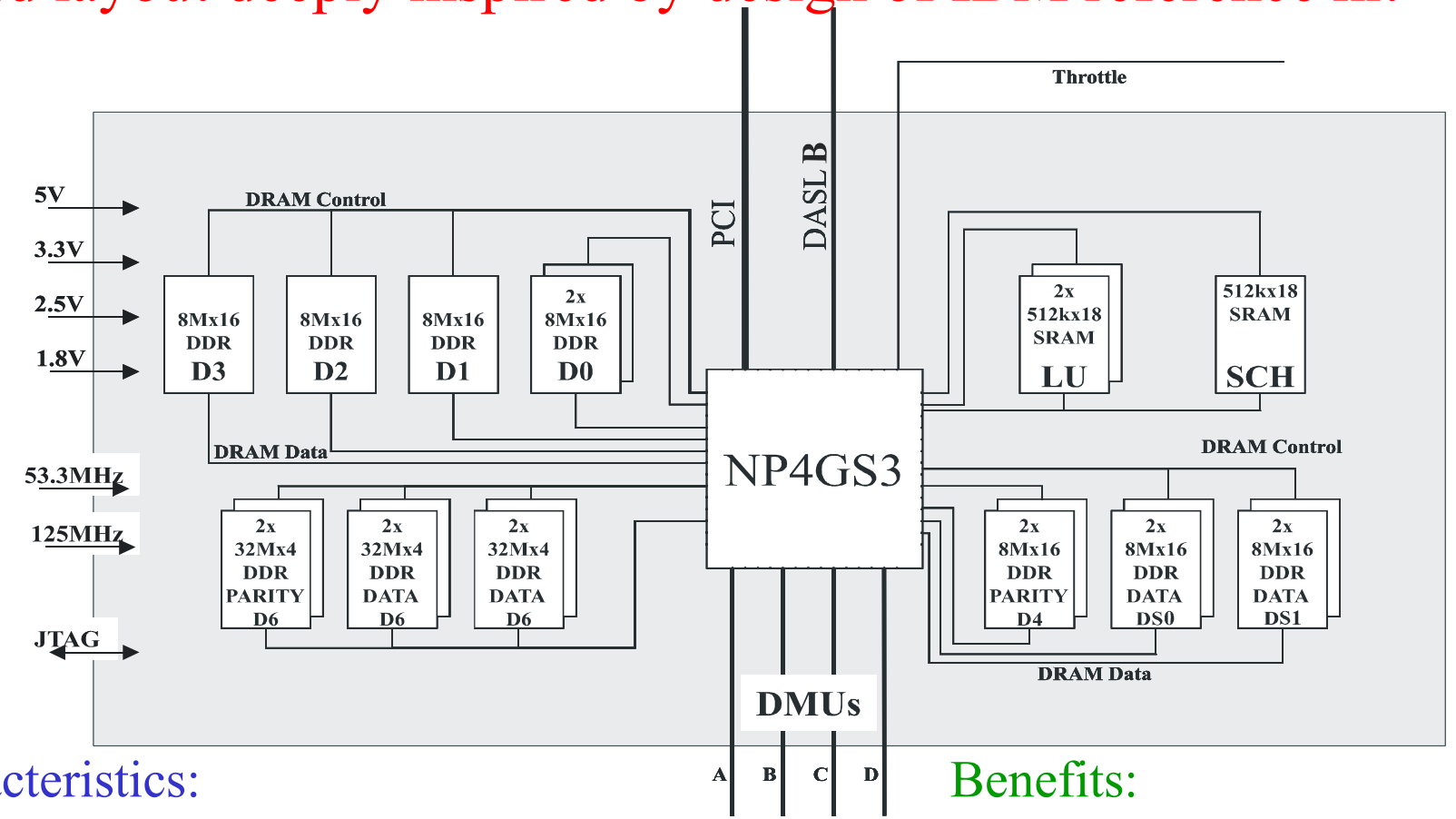
- 9Ux400 mm single width VME-like board (compatible with LHCb standard boards)
- 1 or 2 Mezzanine Cards containing each
  - 1 Network Processor
  - All memory needed for the NP
  - Connections to the external world
    - PCI-bus
    - DASL (switch bus)
    - Connections to physical network layer
    - JTAG, Power and clock
- PHY-connectors
- Trigger-Throttle output
- Power and Clock generation
- LHCb standard ECS interface (CC-PC) with separate Ethernet connection

Architecture



Mezzanine Cards

Board layout deeply inspired by design of IBM reference kit



Characteristics:

- ~14 layer board
- Constraints concerning impedances/trace lengths have to be met

Benefits:

- Most complex parts confined
- Much fewer I/O pins (~300 compared to >1000 of the NP
- Modularity of overall board

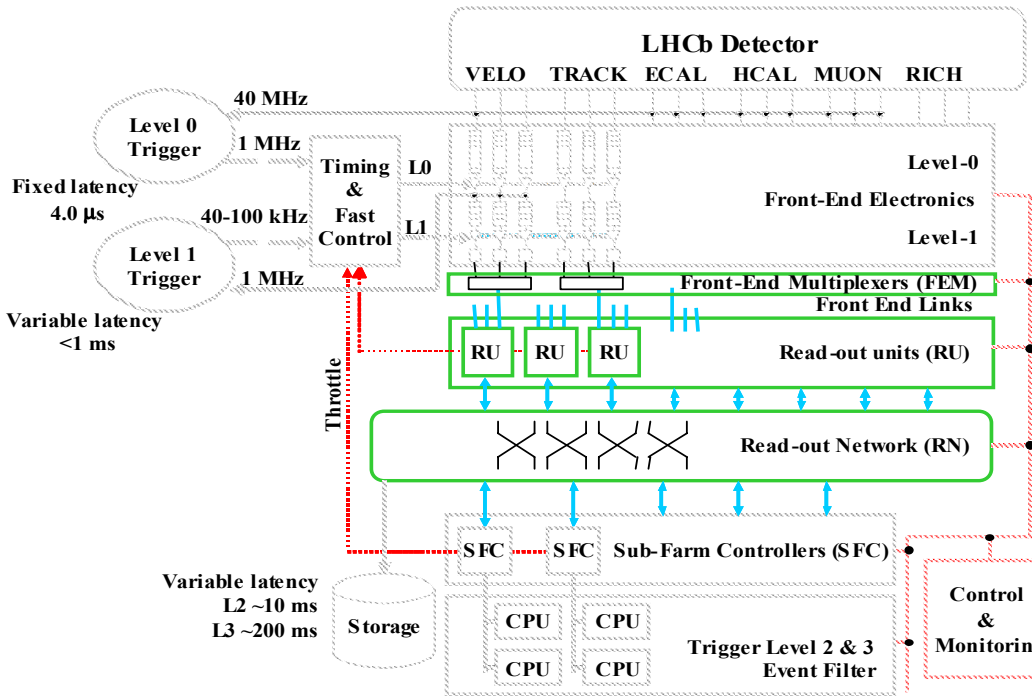
Features of the NP-based Module

- The module outlined is completely generic, i.e. there is no a-priori bias towards an application.
- The software running on the NP determines the function performed
- Architecturally it consists just of 8, fully connected, Gb Ethernet ports
- Using GbEthernet implies
  - Bias towards usage of Gb Ethernet in the Readout network
  - Consequently needs Gb Ethernet-based S-Link interface for L1 electronics (being worked-on in Atlas)
  - No need for NICs in Readout Unit (availability/form-factor)
- Gb Ethernet allows to connect at any point in the data-flow a few PCs with GbE interfaces to debug/test

Applicability in LHCb

Applications in LHCb can be

- DAQ
  - Front-End Multiplexing (FEM)
  - Readout Unit
  - Building Block for switching network
  - Final Event-Building Element before SFC
- Level-1 Trigger
  - Readout Unit
  - Final Event-Building stage for Level-1 trigger
  - SFC functionality for Level-1
  - Building block for event-building network



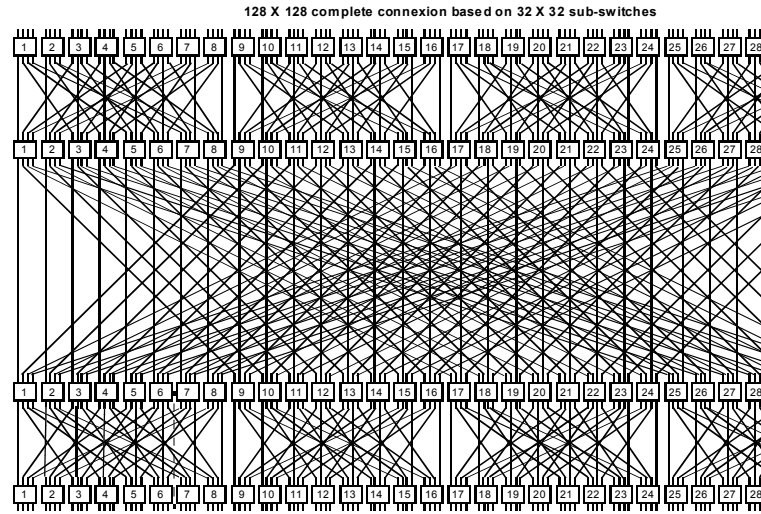
(see later)

DAQ - FEM/RU Application

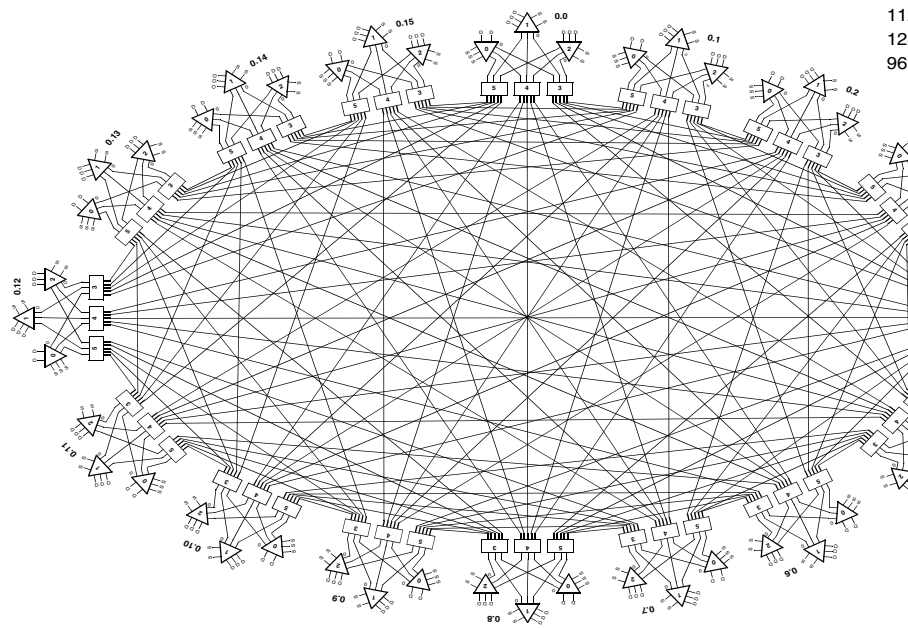
- FEM and RU applications are equivalent
- The NP-Module allows for any multiplexing N:M with N + M ≦ 8 (no de-multiplexing!), e.g.
  - ➢ N:1 data merging
  - ➢ Two times 3:1 if rate/data volumes increase or to save modules (subject to partitioning of course)
- Performance good enough for envisaged trigger rates (≦100 kHz) and any multiplexing configuration (Niko's presentation)

DAQ - Event-Building Network

- NP-Module is intrinsically an 8-port switch.
- Can build any sized network with 8-port switching element, e.g.
  - Brute-force Banyan topology, e.g. 128x128 switching network using 128 8-port modules



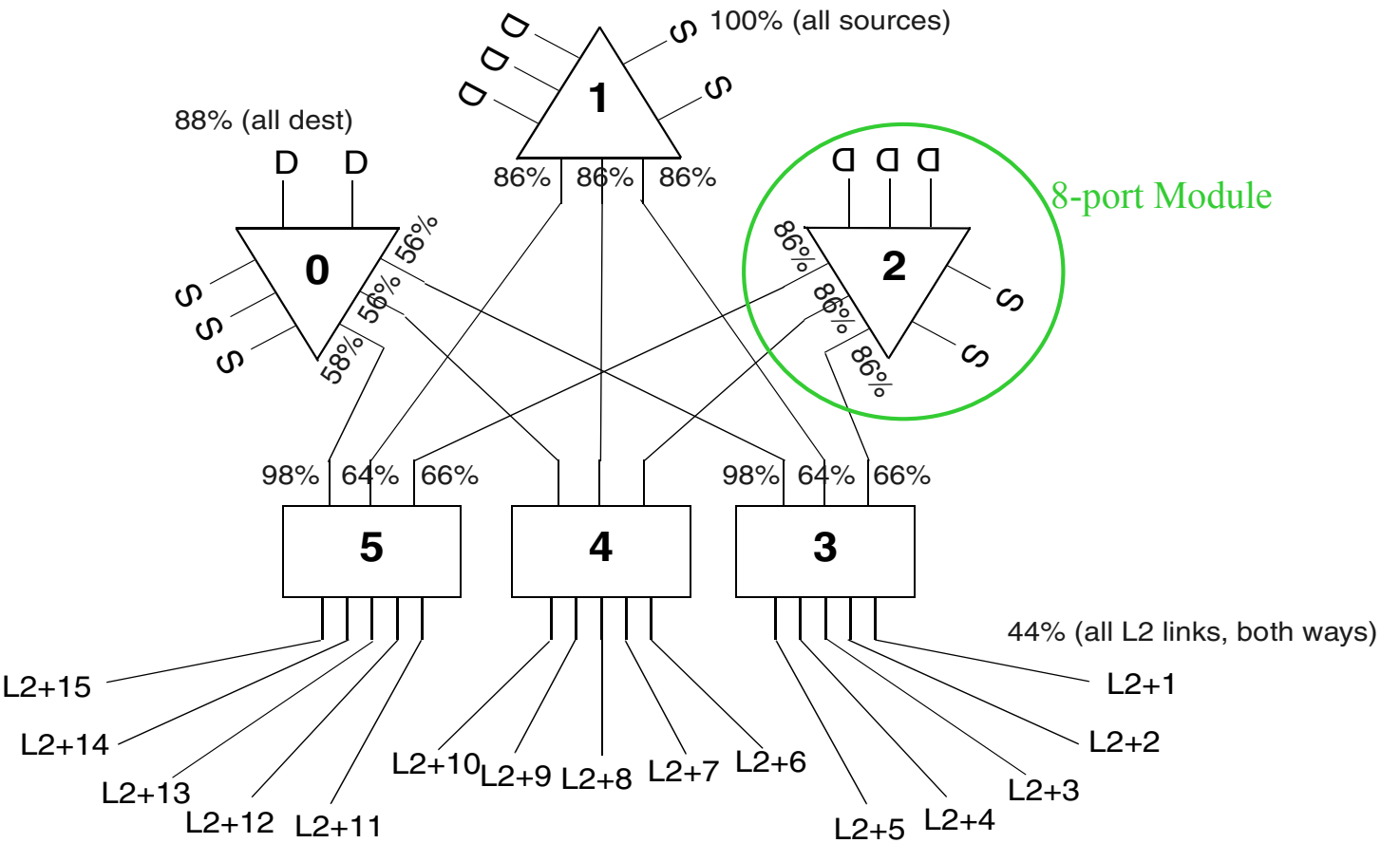
  - More elaborate topology, taking into account special traffic pattern (~unidirectional), e.g. 112x128 port topology using 96 8-port modules



Benefits:

- Full control over and knowledge of switching process (Jumbo Frames)
- Full control over flow-control
- Full Monitoring capabilities (CC-PC/ECS)

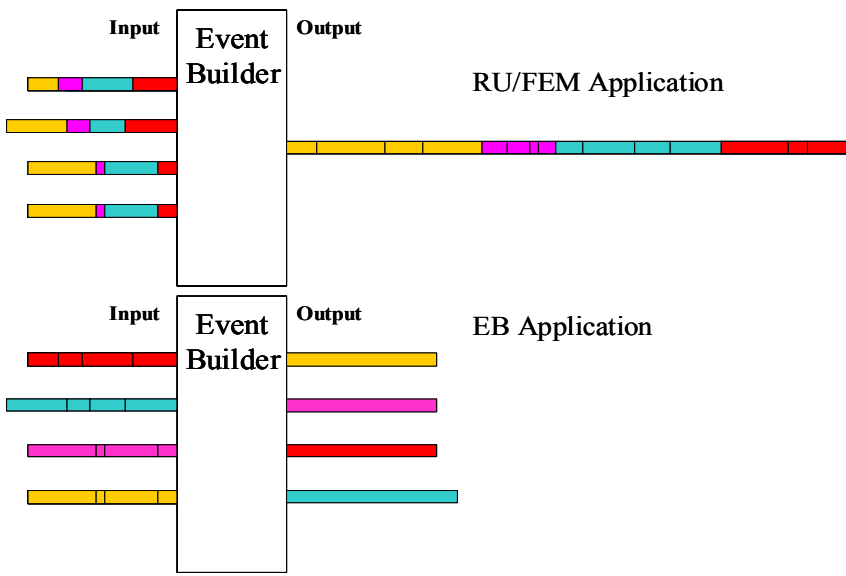
Event-Building Network - Basic Structure



DAQ - Final Event-Building Stage (I)

- Up to now the baseline is to use "smart NICs" inside the SFCs to do the final event-building.
  - Off-load SFC CPUs from handling individual fragments
  - No fundamental problem (performance sufficient)
  - Question is future directions and availability.
    - Market is going more towards ASICs implementing TCP/IP directly in hardware.
    - Freely programmable devices more geared for TCP/IP (small buffers)

- NP-based Module could be a replacement
  - 4:4 Multiplexer/Data Merger

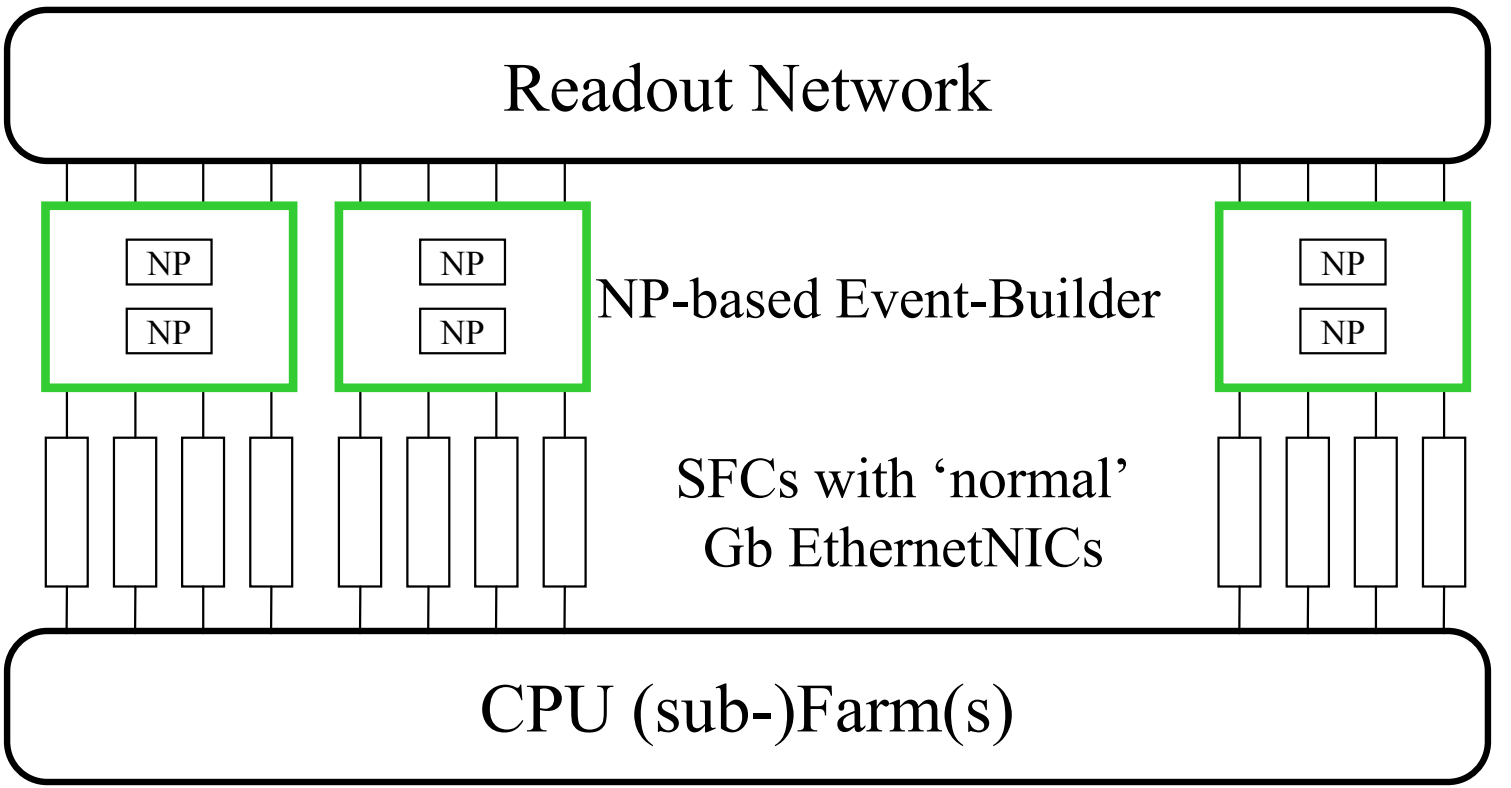


Only a question of the software loaded

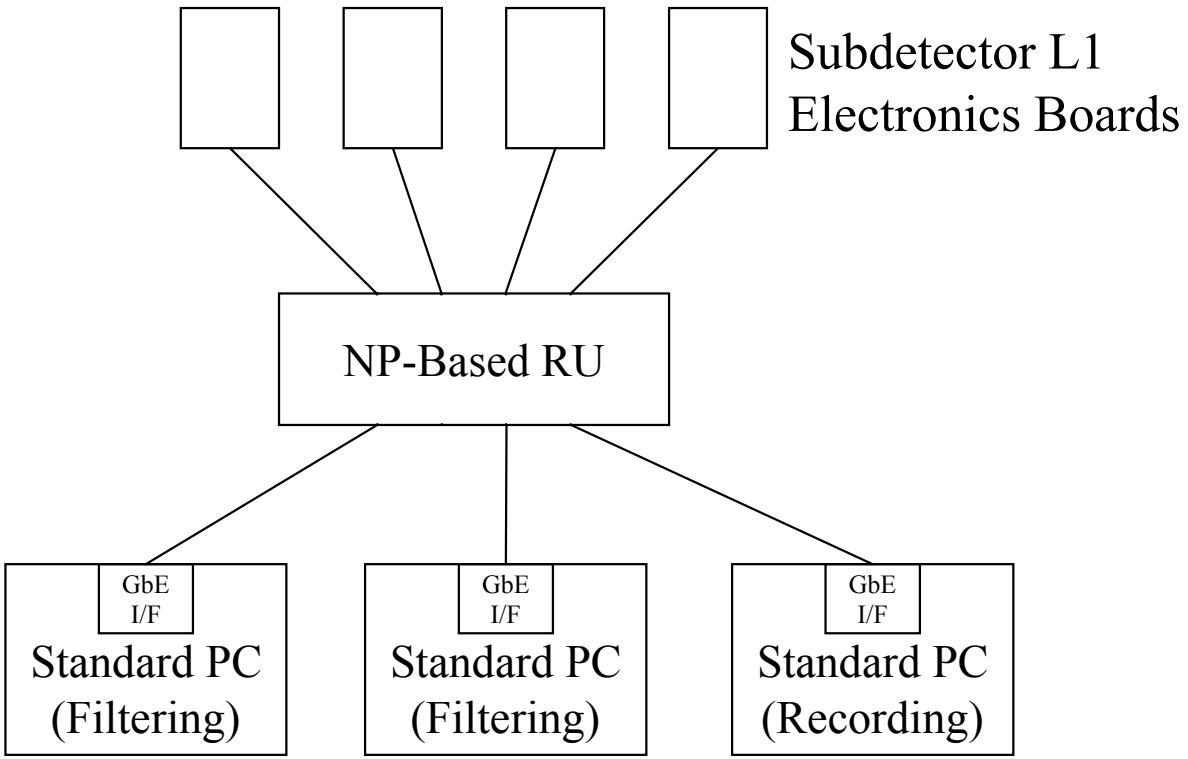
Actually the software written so far doesn't know about ports in the module

Final Event-Building Stage (II)

- Same generic hardware module
- ~Same software if separate layer in the dataflow
- SFCs act 'only' as big buffers and for elaborated load balancing among the CPUs of a sub-farm



Example of small-scale Lab Setup



Centrally provided:

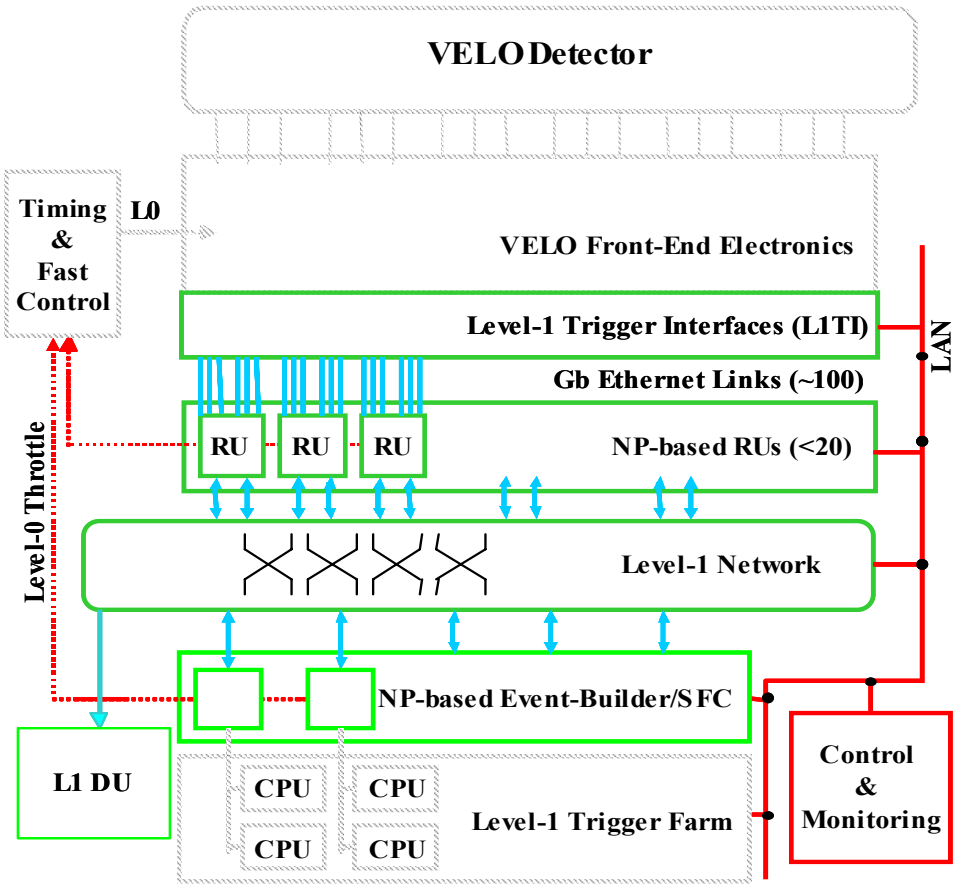
- Code Running on NP to d
  event-building
- Basic framework for
  filter nodes
- Basic tools for recordin
- Configuration/Control/M
  nitoring through ECS

Level-1 Trigger Application (Proposal)

Basically exactly the same as for the DAQ

- Problem is structurally the same, but different environment (1.1 MHz Trigger rate and small fragments)
- Same basic architecture
- NP-RU module run in 2x3:1 mode
- NP-RU module for final event-building (as in DAQ) and implementing SFC functionality (load-balancing, buffering)

Performance sufficient! (see Niko's presentation)



Design and Production

- Design
  - In principle a 'reference design' should be available from IBM
  - Based on this the Mezzanine cards could be designed
  - The mother-board would be a separate effort
  - Design effort will need to be found
    - inside Cern (nominally "cheap")
    - Commercial (less cheap)
  - Before prototypes are made, design review with IBM engineers and extensive simulation performed
- Production
  - Mass production clearly commercial (external to Cern)
  - Basic tests (visual inspection, short/connection tests) by manufacturer
  - Functional testing by manufacturer with tools provided by Cern (LHCb)
  - Acceptance tests by LHCb

Cost (very much estimated)

- Mezzanine Board
  - Tentative offer of 3 k$/card (100 cards), probably lower for more cards. -> 6 k$/RU
  - Cost basically driven by cost of NP (goes down as NP price goes down)
    - ~1400 $ today, single quantities
    - ~1000 $ in 2002 for 100-500 pieces
    - ~500 $ in 2002 for 10000+ pieces
    - 2003???
- Carrier Board
  - CC-PC: ~150 $
  - Power/Clock generation: ??? (but cannot be very expensive?)
  - Network PHYs (GbE Optical small form-factor): 8x90$
  - Overall: ~2000 $?
- Total: <~8000$ (100 Modules, very much depending on volume)
- Atlas has shown some interest in using the NP4GS3 and also in our board architecture, in particular the Mezzanine card (volume!)

Number of NP-based Modules

DAQ				
		Type		Installed Bandwidth
FEM	50	8-port		
RU	90	8-port		11.25 GB/s
Readout Network	96	8-port		14 GB/s
Event-Builder	23	8-port		
Total Units	259			
Cost [$]			2072000	
only FEM/RU	140			
Cost [$]			1120000	

Level-1				
				installed Bandwidth
FEM				
RU	32	8-port		8 GB/s
Readout Network	48	8-port		
Event-Builder				
Total Units	80			
Cost [$]			640000	
only FEM/RU	32			
Cost [$]			256000	

Notes:

- For FEM and RU purposes it is more cost effective to use the N based RU module in a 3:1 multiplexing mode. This reduce the number of physical boards b factor ~1/3
- For Level-1 the number is determined by the speed of the output link. A reduction in the fragment header can lead to a substantial saving. Details to be studied.

Summary of Features of a Software-Driven RU

- Main positive feature is the offered flexibility to new situations
  - Changes in running conditions
  - Traffic shaping strategies
  - Changes in destination assignment strategies
  - Etc…
- but also elaborate possibilities of diagnostic and debugging
  - Can put debug code to catch intermittent problems
  - Can send debug information via the embedded PPC to the ECS
  - Can debug the code or malfunctioning partners in-situ

Summary (I) - General

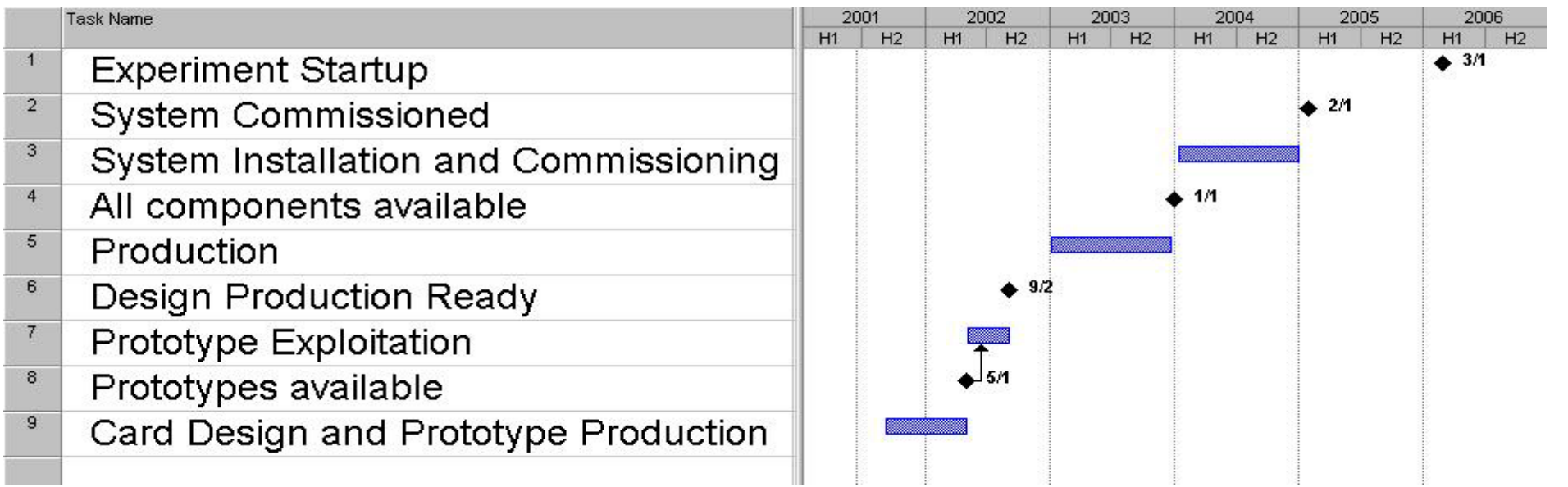
- NP-based RU fulfils the requirement in speed and functionality
- There is not yet a detailed design of the final hardware available, however a functionally equivalent reference kit from IBM has been used to prove the functionality and performance.

Summary (II) - Features

- Simulations show that performance is largely sufficient for all applications
- Measurements confirm accuracy of simulation results
- Supported features:
  - Any network-based (Ethernet) readout protocol is supported (just software!)
  - For all practical purposes wire-speed event-building rates can be achieved.
  - To cope with network congestion 64 MB of output buffer available
  - Error detection and reporting, flow control
    - 32-bit CRC per frame
    - Hardware support for CRC over any area of a frame (e.g. over transport header). Software defined.
    - Embedded PPC + CC-PC allow for efficient monitoring and exception handling/recovery/diagnostics
    - Break-points and single stepping via the CC-PC for remote in-situ debugging of problems
  - At any point in the dataflow standard PCs can be attached for diagnostic purposes

Summary (III) - Planning

- Potential future work programme
  - Hardware: It's-a-depends-a… (external design: ~300 k$ design+production tools)
  - ~1 m•y of effort for infrastructure software on CC-PC etc. (test/diagnostic software, configuration, monitoring, etc.)
  - Online team will be responsible for deployment, commissioning and operation, including Picocode on NP.
- Planning for module production, testing, commissioning (depends on LHC schedule)



Summary (IV) – Environment and Cost

- Board: aim for single width 9Ux400 mm VME, power requirement: ~60 W, forced cooling required.
- Production Cost
  - Strongly dependant on component cost (later purchase→ lower price)
  - In today's prices (100 Modules):
    - Mezzanine card: 3000 $/card (NB: NP enters with 1400$)
    - Carrier card : ~2000 $ (fully equipped with PHYs, perhaps pluggable?)
    - Total: ~8000 $/RU (~5000 $ if only one mezzanine card mounted)

Conclusion

- NPs are a very promising technology even for our applications
- Performance is sufficient for all applications and software flexibility allows for new applications, e.g. implementing the readout network and the final event-building stage.
- Cost is currently high, but not prohibitive and is expected to drop significantly with new generations of NPs (supporting 10 Gb Ethernet) entering the scene.
- Strong points are (software) flexibility, extensive support for diagnostics and wide range of possible applications
  →One and only one module type for all applications in LHCb