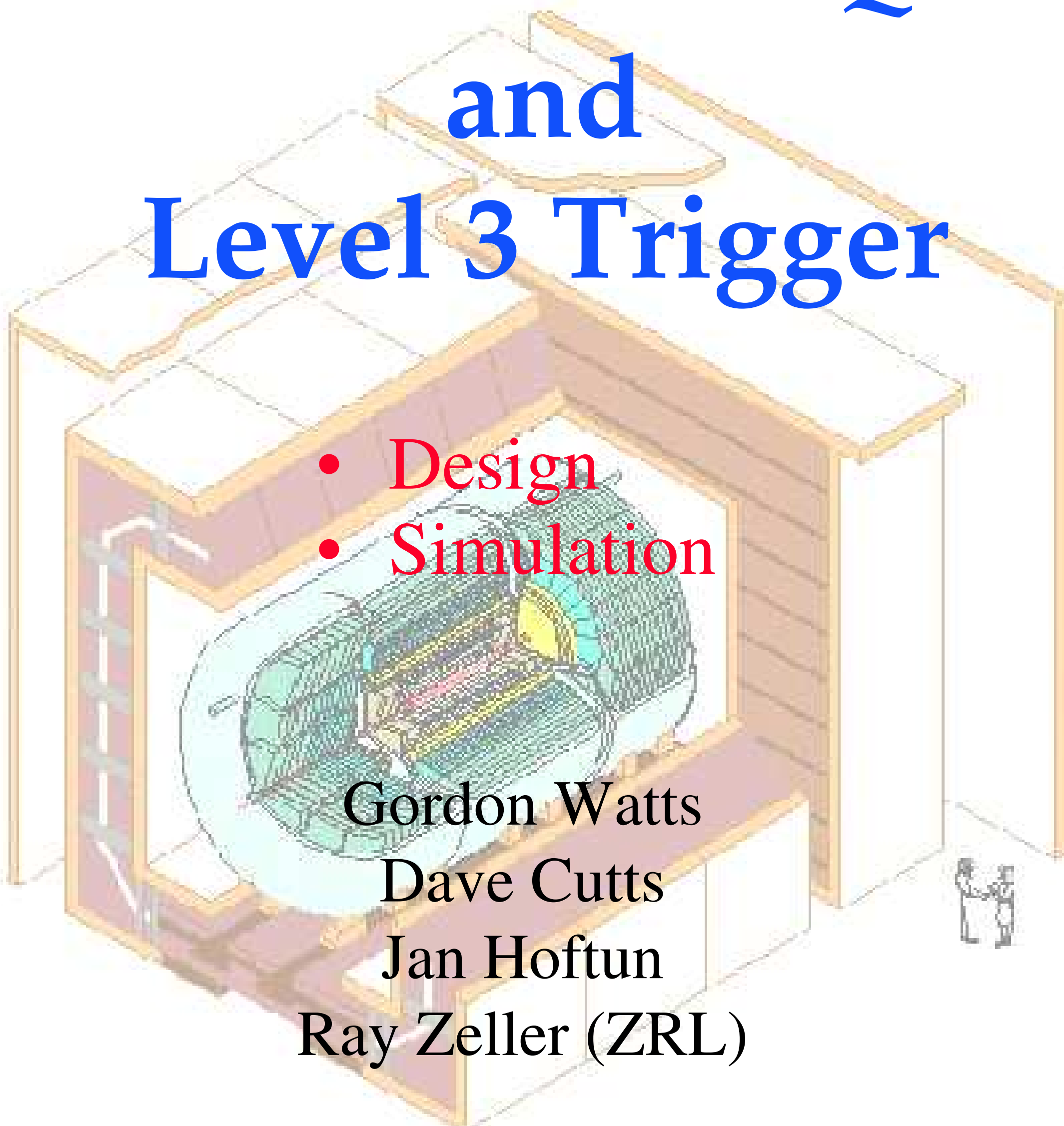
The DØ DAQ and Level 3 Trigger



- Design
- Simulation

Gordon Watts
Dave Cutts
Jan Hoftun
Ray Zeller (ZRL)

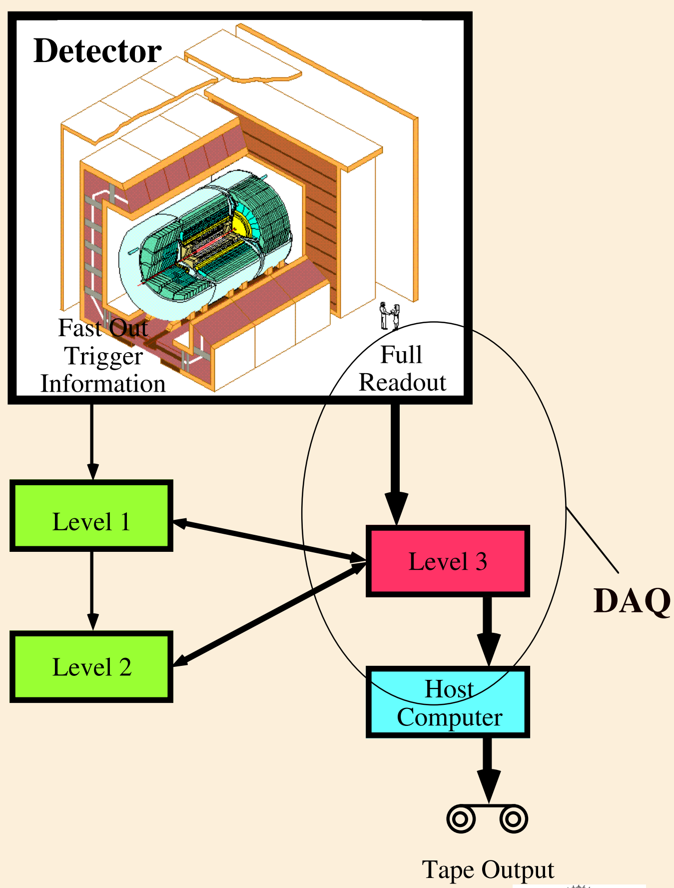
IEEE Real Time '97
Beaune, France

Brown High Energy Physics

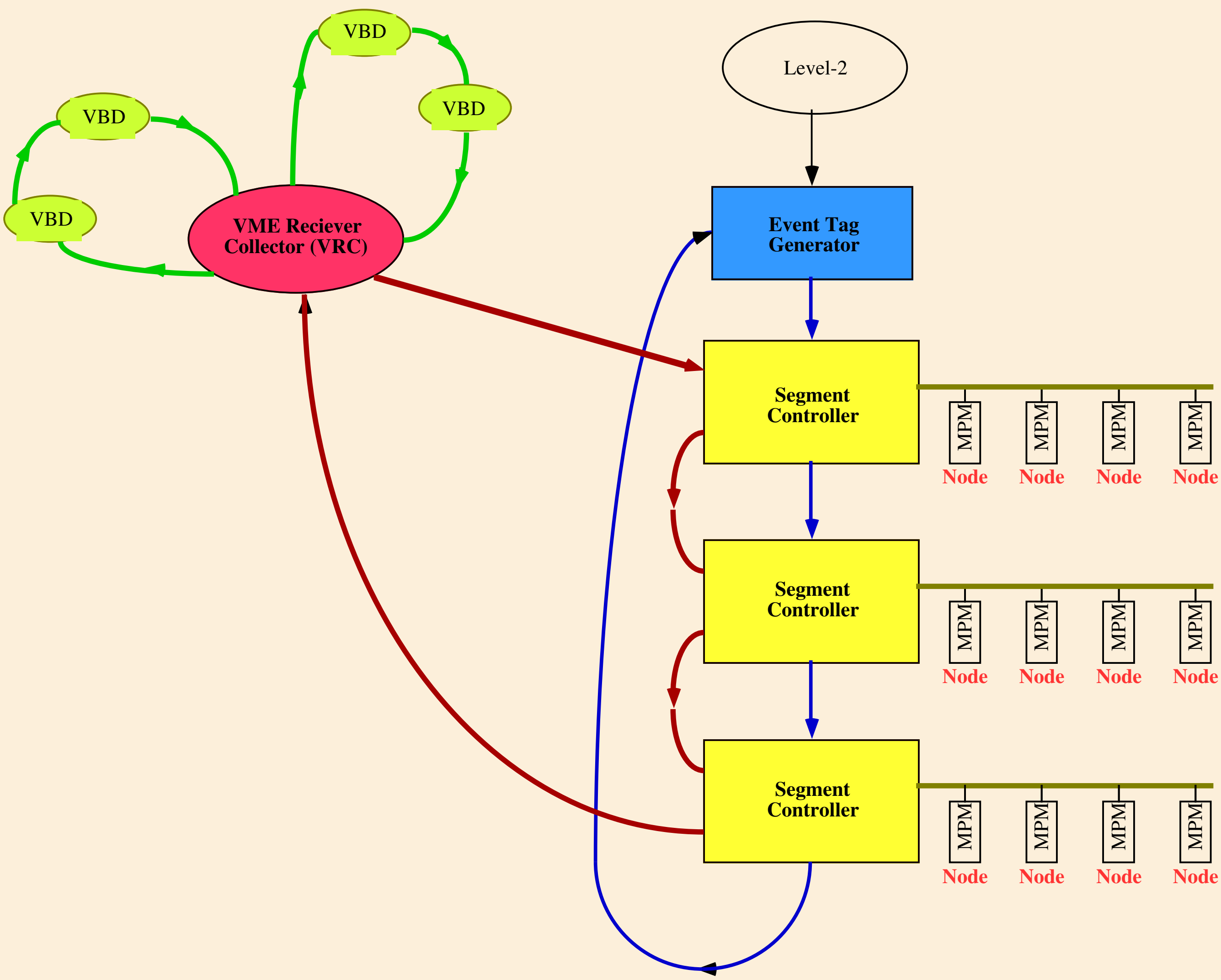
ZRL

Introduction

- DAQ
  - Data from the Front End Crates of the D∅ detector.
  - Data from one event is sent to one farm node
  - Triggered events are sent to the host system where they are recorded on tape
- Trigger
  - Events are checksummed, etc.
  - Physics algorithms (optimized for speed)!



The Datapaths



The Architecture

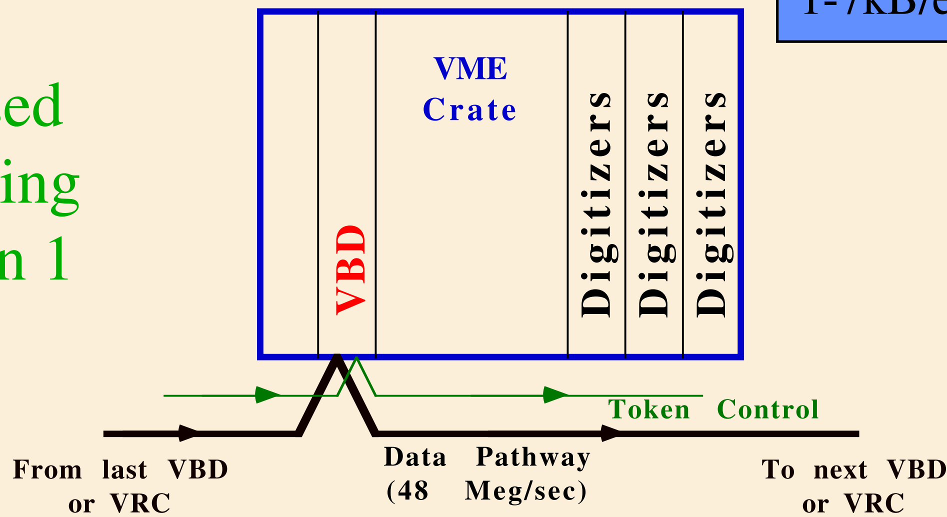
- Data transport based on custom architecture.
  - Does not use standard network protocals (like ATM or IP)
  - Does use a commodity physical layer: Fibre Channel.
- Data flow is unidirectional.
- Each packet is sent only once
  - no retransmission in case of errors.
- System causes deadtime when it is full.
  - Does not loose packets.
- Routing of data is driven by the contents of the data packet.
  - i.e. Sender does not know or care where data is going.
- Physical transport layer "packet" consitst of the entire data block from the VBD
  - "packets" are never split by the transport layer.

Front End Crates

- Digitized data enters DAQ system via a VBD.
  - Typically 1-7kB data per event per FEC.
- VBD
  - One in each Front End Crate (FEC).
  - Built ≈ 10 years ago.
  - Several VBDs drive a 48 MB/sec readout path
    - Syncronization is done via a Token.
  - List-based processor accesses VME space
  - Dual 256kB buffers
  - Adds header, trailer, and checksum to data.
  - Single 9U VME Card

1-7kB/event/FEC

Used during Run 1

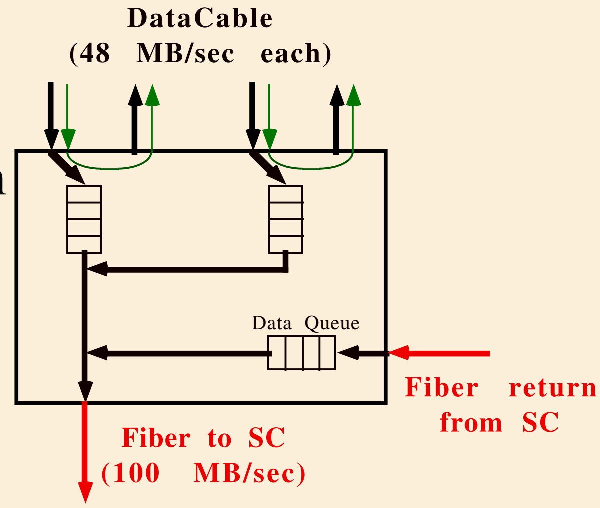


VRCs

- Drives the token loop that sequences the readout of a group of VBDs.
- Designed so that data does not sit in VRC; passed along immediately.
- Suspend VBD readout if no room.
- Input:
  - 2 48 MB/s datacables (from VBDs)
  - 1 100 MB/s Fibre Channel for overflow.

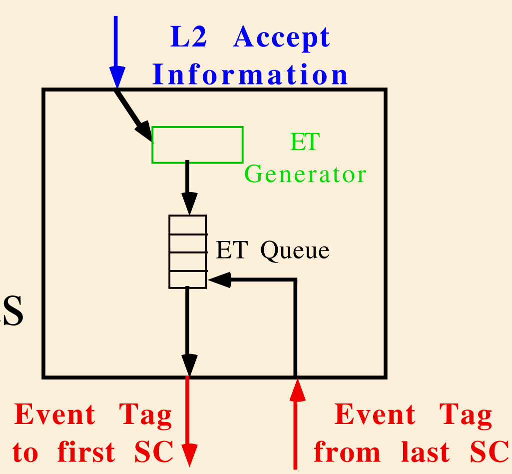
Actual use very small

- Output:
  - 1 100 MB/s Fibre Channel to Segement Controller
- Dual EuroCard format
  - PC Motherboard for control, second custom hardware card.



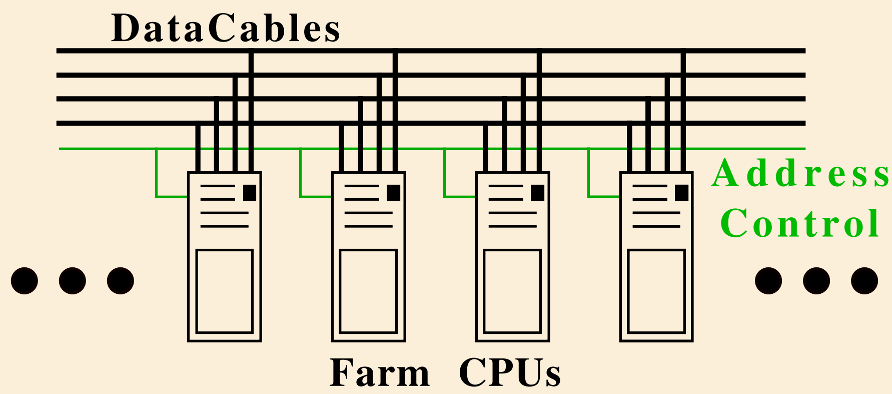
Event Tag Generator

- L2 sends to the L3 ETG:
  - L2 trigger bits (128)
  - Event number
- ETG then
  - Sorts trigger by class of event (calib, regular, etc.)
  - Creates Event Tag with class, number, crate readout list
  - Circulates ET to all Segment Controlers
- When a SC is done, it returns the ET to the ETG.
- ETG is where most of our monitor information is accumulated
- Can send diagnostic packets to SCs
- Crate of VME and custom electronics
  - Includes a PC for control and monitoring.



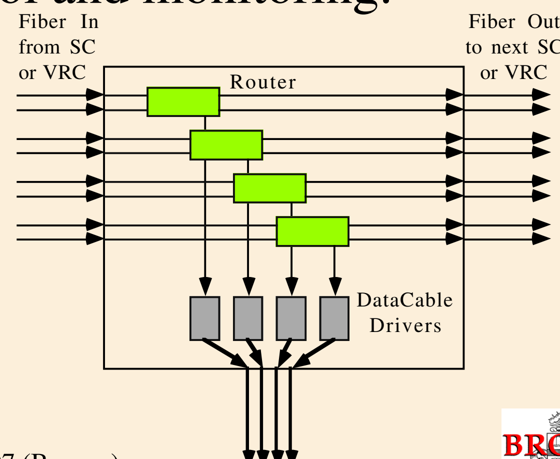
Farm Segment

- Each segment has up to 16 computers
  - 4 or 6 DC per segment, running in parallel
- Each SMP CPU is connected to the datacables via a VME crate and multiport memory
- Each CPU informs SC when it is ready to accept an event.
- Readout control/latching is done by the SC.



Segment Controllers

- One Fiber input per VRC.
- Route required data blocks from fibers to farm segments
  - SC manages the DataCables in a Farm Segment
- Maintains routing table to do the routing
  - Updated via the ETs from the ETG
- Passes blocks it doesn't know about or doesn't have room for on fiber output.
  - To next SC or back to VRC (recirculation)
- Data buffer to queue data blocks going down into nodes.
- VME sized crate and custom rack mount.
  - PC included for control and monitoring.



Recirculation

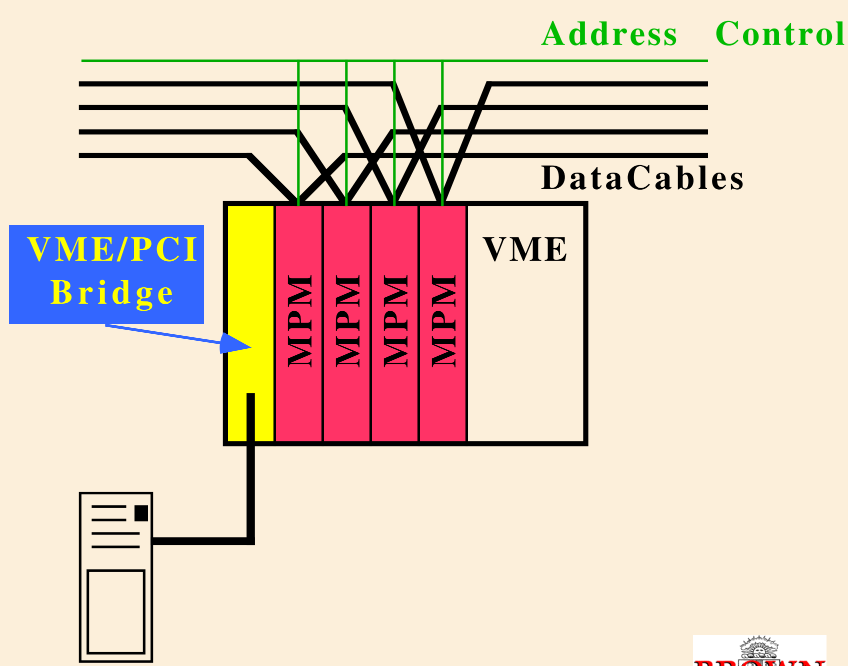
- Data blocks a SC doesn't know what to do with or have room for are passed onto the next SC.
- Last SC in daisy chain returns blocks to the VRC they started from.
- The VRC retransmits the returned blocks right away.
- rate self-limiting mechanism
  - The more blocks recirculated, the less room for new blocks to enter system at VRC
  - Monitor systems can observe retransmission.
- A block is deleted if it is retransmitted too many times
  - Count is kept in header of block
- SC times-out events that aren't completed.

Farm CPUs

- Intel PIIs or other cheap comodity processors
  - SMP systems, quad-processor most likely.
- Running Windows NT
  - Good OS services.
  - Good development support (debugging) and efficient compilers.
  - Code will be based on our offine framework.
  - Augment with efficient algorithms
- 48 nodes must render a decision every 48 miliseconds (1000 Hz).
  - 4 processors per node, each one event at a time
    - factor of 4 longer (whew!!).
    - Simulation shows it is really a factor of 3.

Farm CPUs

- PCI/VME Bridge with DMA capabilities to shuttle data in and out of node.
- Upon event arrival (from SC), interupt given to PC
  - DMA transfer starts
- When node is ready for next event, it notifies the SC.
- Accepted events are pumped out along second data cable (48 MB/sec) to collector node.



Staging

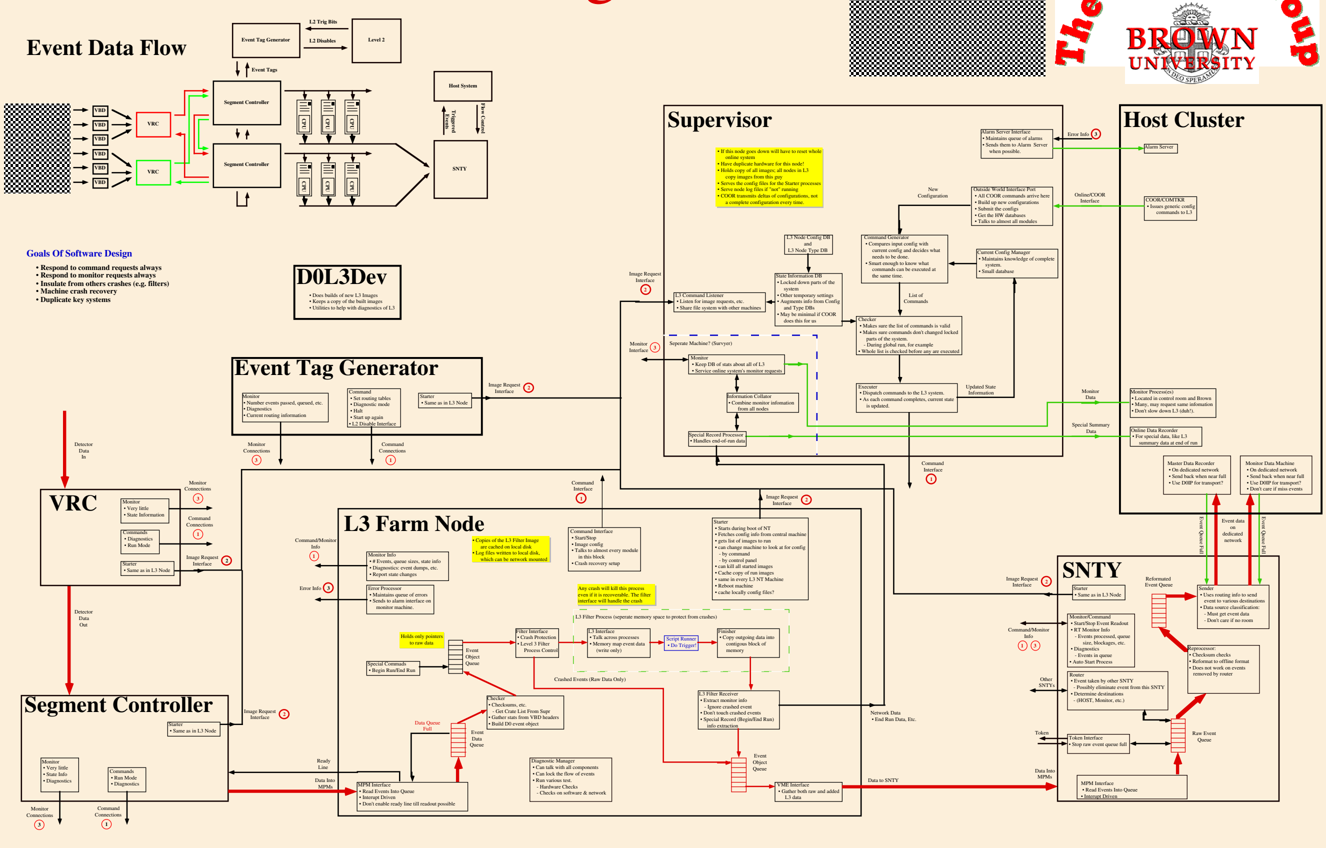
- Designed to expand
  - From Run II to TeV33 era.
- Number of fiber links/VRCs can be expanded to 48
- Number of farm segments to 16
- Bandwith to farm segments to 288 MB/sec

	Stage A	Stage B	Stage C	Stage D	Stage E
# VRCs	8	8	8	12	24
# SCs	2	3	4	4	8
# L3 Nodes	48	48	48	64	64
Bandwidth (MB/s)	400	600	800	1200	2400
50% Event Rate (Hz)	0.8	1.2	1.6	2.4	4.8

Software

- Huge project; topic at next CHEP in Sept ‘98

Level 3 Software Design

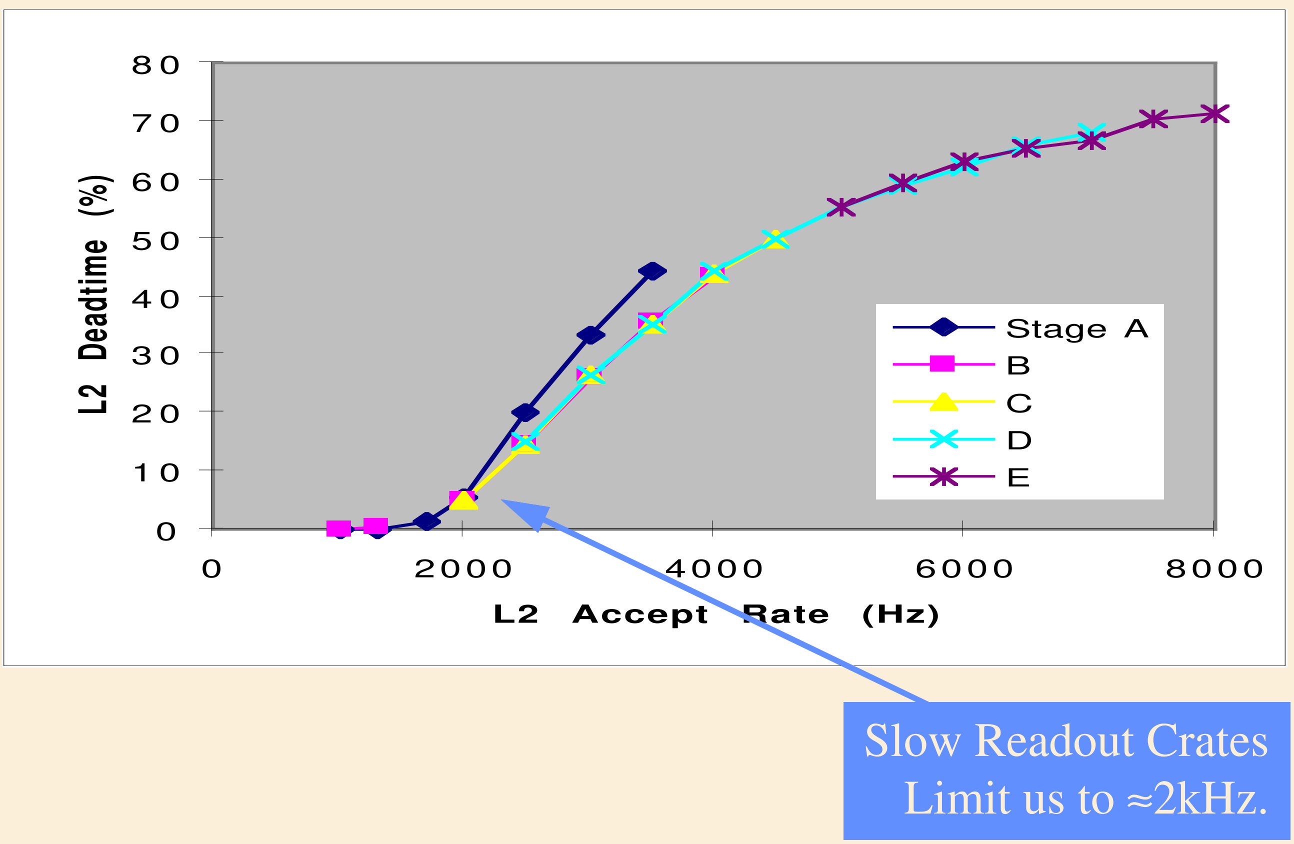


Simulation

- Complex system; must understand behavoir before construction
- Simulation gives feedback on the design
  - Can predict how likely it is a design choice will work.
- Use a descrete event simulation.
  - Not a queuing simulation
    - Closer to VHDL.
  - Ex: VBD
    - 1 thread handles the VME backplane
    - 1 thread handles the datacable front-end
    - Communicate between with the dual buffers.
- MODSIM III (CACI).
  - Includes nice graphics
  - http://www.caciasl.com/
- 250K lines of source code!

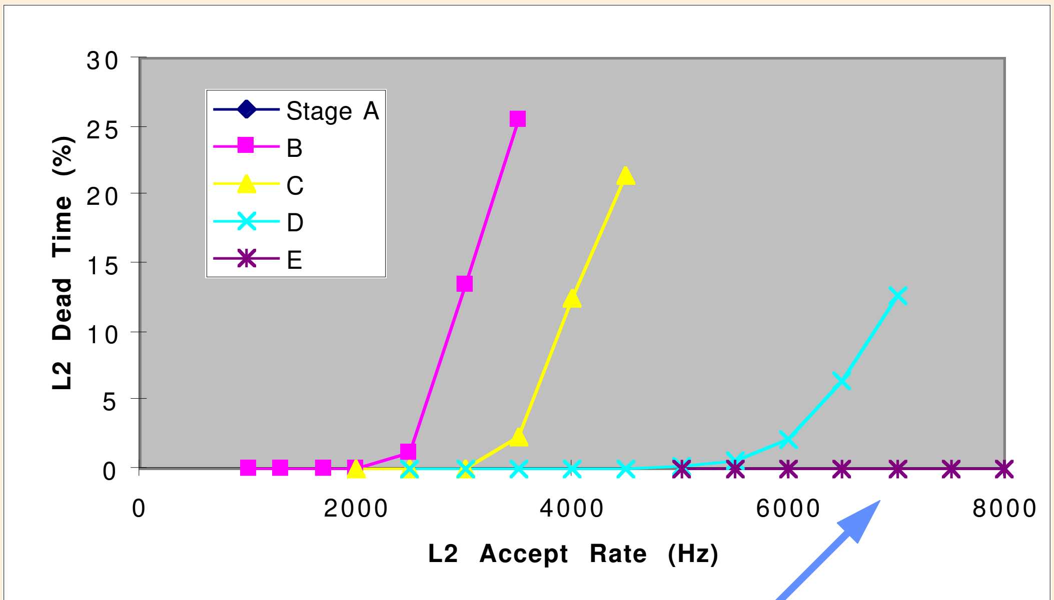
Event Rate

- First question: Will it go as fast as we think?
- Complex answer:
  - The DAQ must be fed at a high rate or the various stages don't make a difference.



Event Rate

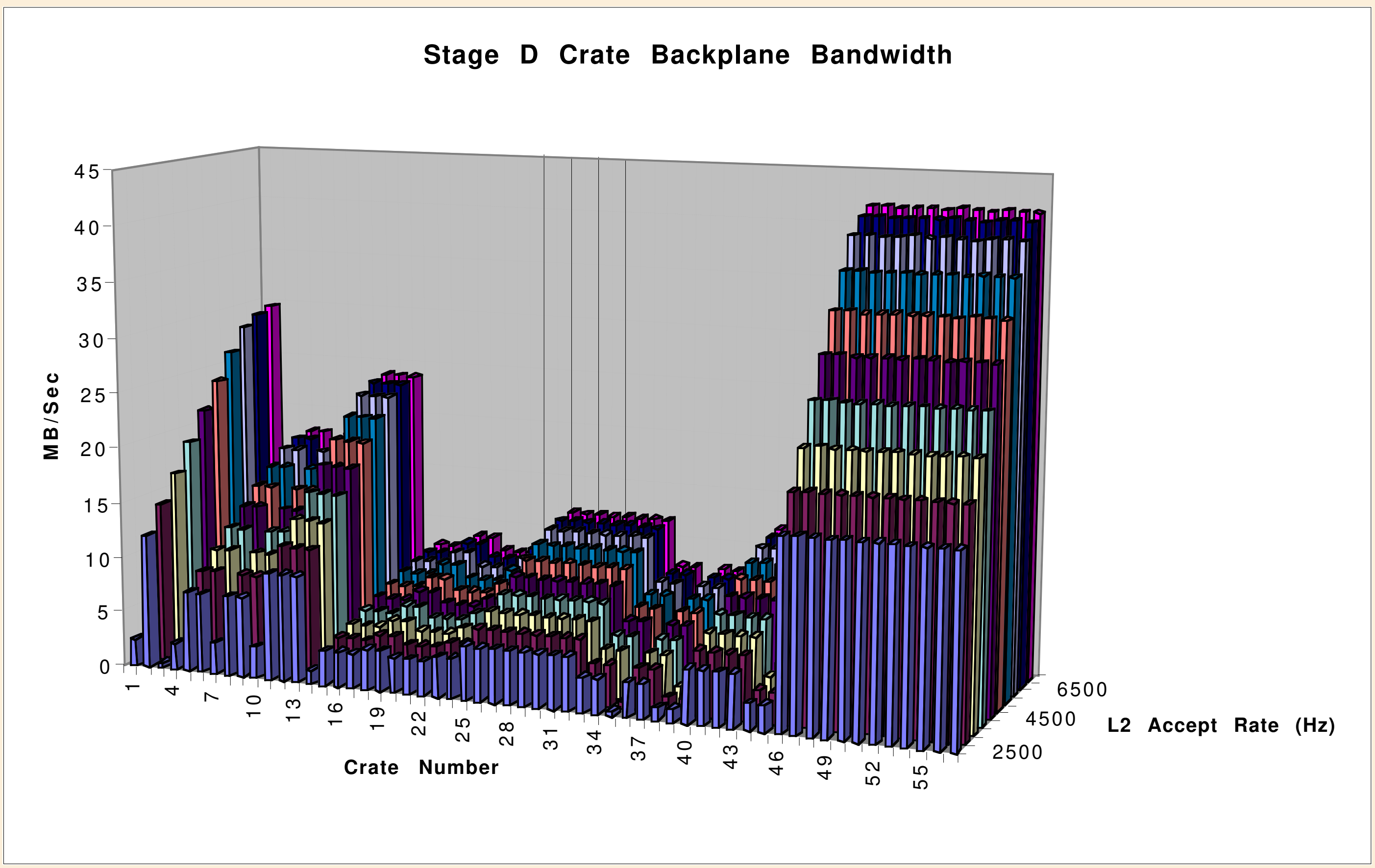
- Remove those restrictions
  - Unreal VME backplane speeds for some of the very high trigger rates.



Goes to >10 kHz before real deadtime appears

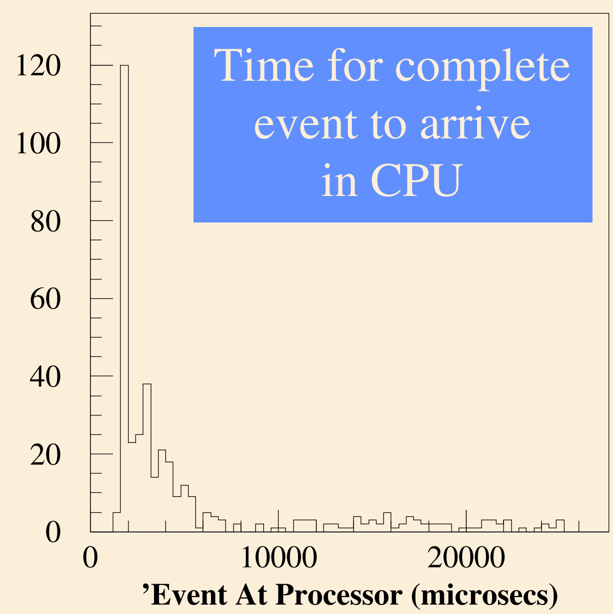
VME Backplane Speed

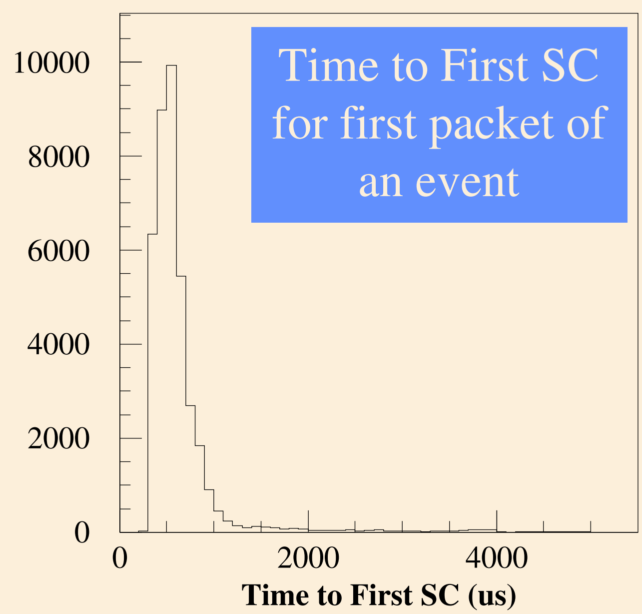
- Can look in detail at what is going on in the system
  - set of ntuples is written out, and then processed.



Time to ETG

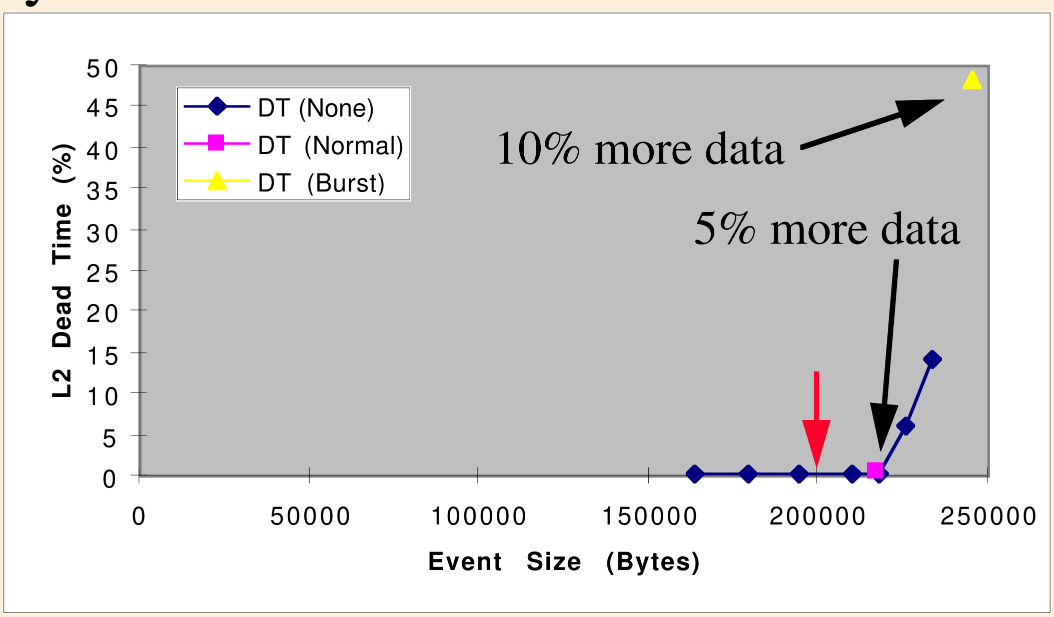
- For best efficiency SC must know about event before it arrives.
  - Prevent uneeded recirculation.
- This defines how long the ETG has to circulate the ETs.





Bursty Events

- At D∅ we have events where large parts of a detector light up
  – The muon system after a splash or other background effect
- We can simulate the effect to make sure the DAQ system is not adversely affected by bursts of data.



- Behaves only as if event size increased.
  - Good!

Conclusions

- A modern architecture that takes advantage of legacy hardware investments.
- A directed data flow system
- Well simulated so suprises after building should be 'smaller.'
  - Not just how fast will system run, but how will it perform under adverse conditions?
  - Other modifications being done
    - Error conditions (dropped data buffers, etc.)
    - Timeouts
    - Small buffer sizes
    - Update timing in simulation as prototypes arrive.