

DAQ Status

Emlyn Corrin

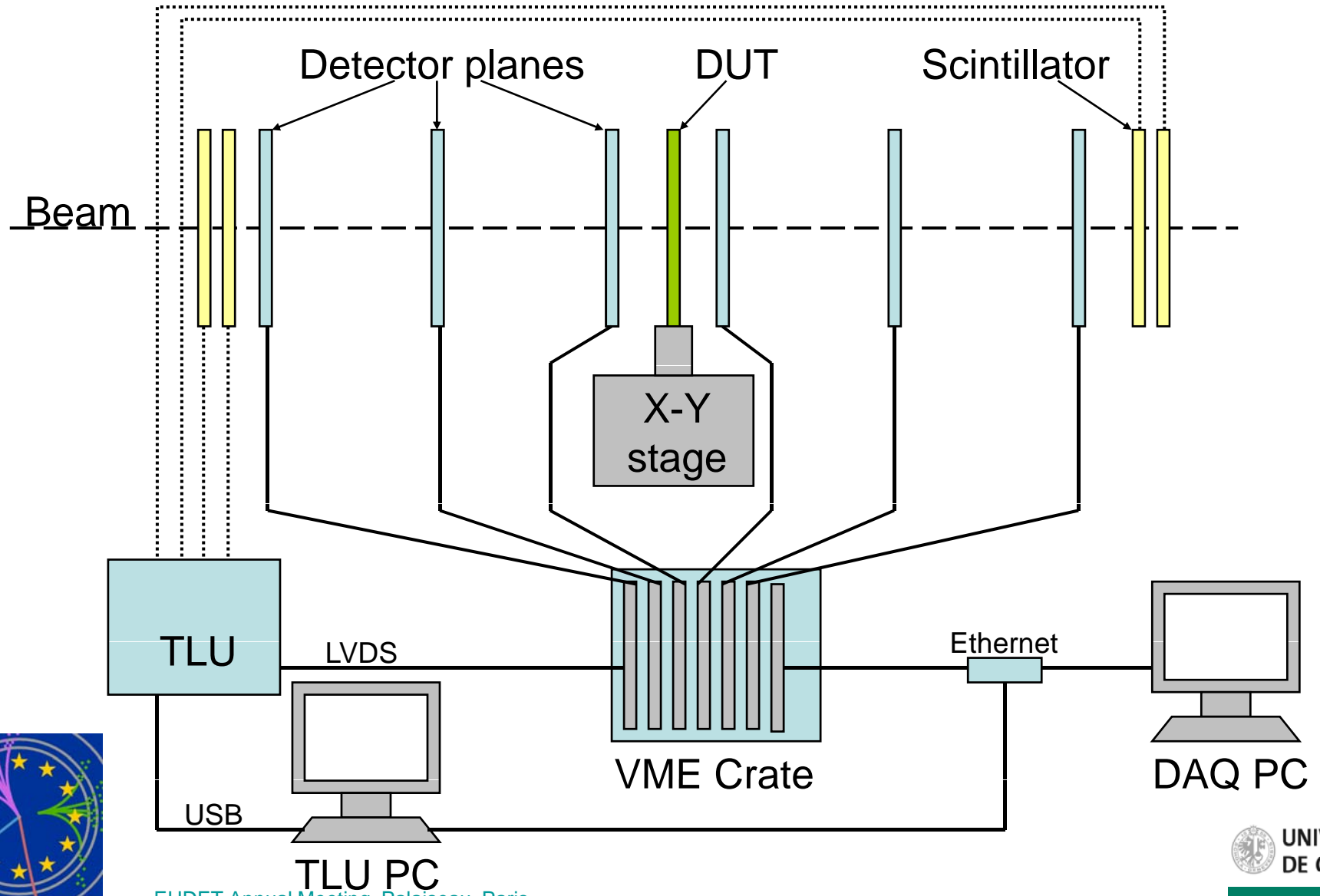


Outline

- Hardware
- Software
 - Architecture
 - Producers
 - Data Collector
 - Run Control
 - Log Collector
 - Root Monitor
- Remaining Issues
- Summary



Hardware



Hardware

- VME Crate with 6 EUDRBs and 1 MVME6100 CPU running Linux
 - Only tested with 5 EUDRBs
- TLU controlled by a Linux PC
- DAQ PC: Mac Pro, Quad Core
 - System disks: 2x400GB RAID 0
 - Data disks 4x1TB RAID 0+1 for data
 - Future: upgrade to Mac OS 10.5 with RAID 1 system disks and RAID 5 data



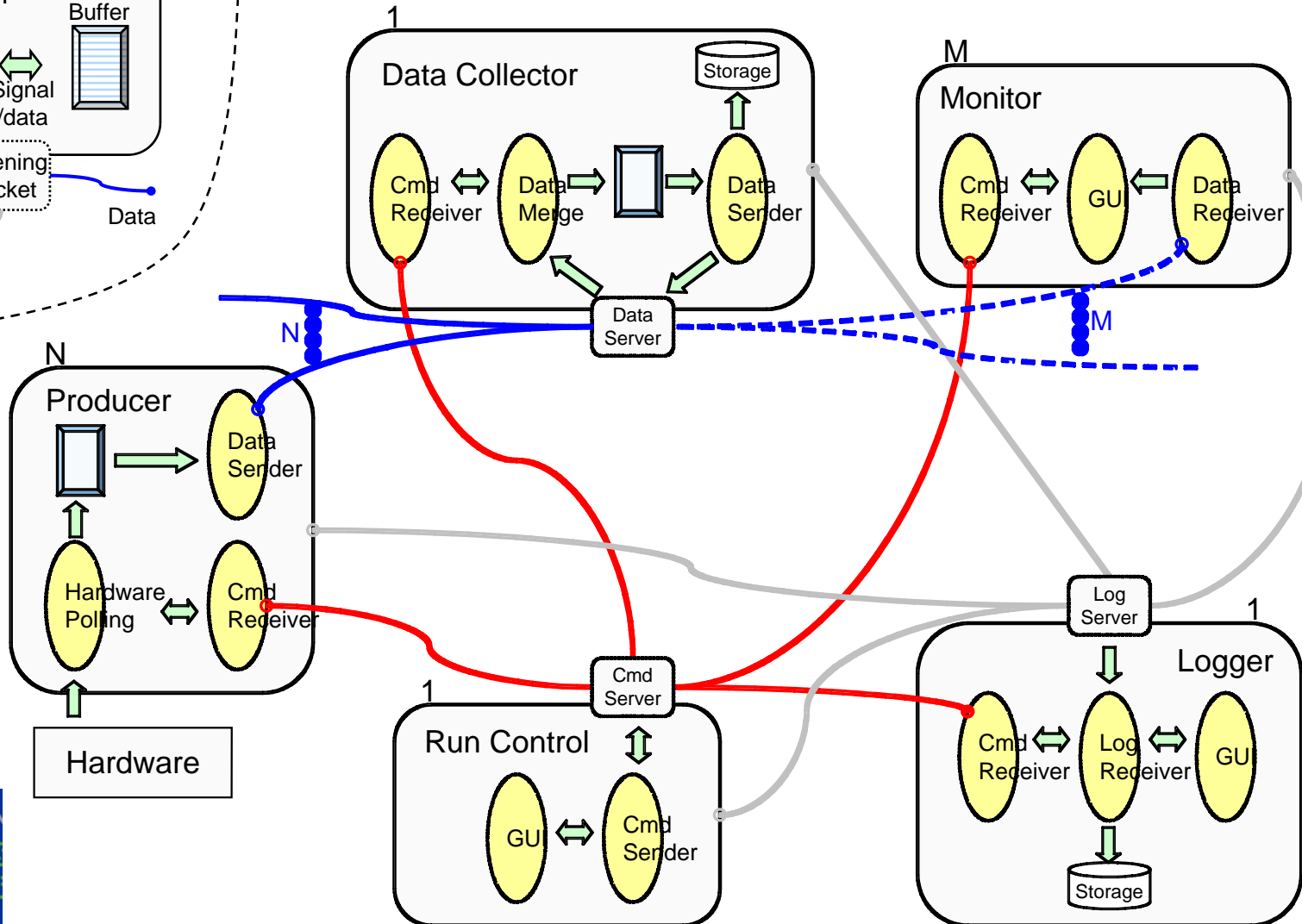
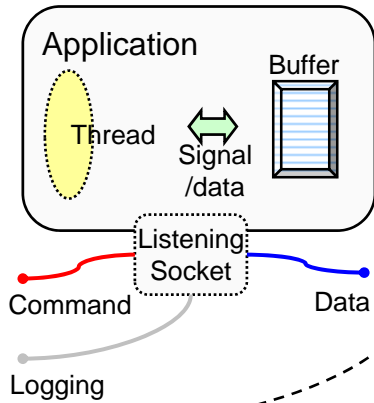
DAQ Software

- Custom DAQ software written in C++
- Uses POSIX for threads and sockets
- Run Control GUI using Qt
- Online Monitor using Root
- Runs on Mac OS X, Linux, and Windows (using cygwin)
- Highly modular, allowing DUTs to be easily integrated into the DAQ
- Software available at: <http://eudet.unige.ch> (or currently <http://eudetmac001.cern.ch>)



Software Architecture

Key:



Producers

- These communicate with the hardware
- They receive commands from Run Control
- They send Events to the Data Collector
- A number of different Producers exist:
 - TestProducer: sends dummy events under user control
 - TLUProducer: sends trigger number and timestamp information
 - EUDRBProducer: reads out EUDRBs and sends data in raw or zero-suppressed modes
 - DEPFETProducer: combination of DEPFET FileWriter with a Producer (receives data from DEPFET DAQ via TCP/IP)



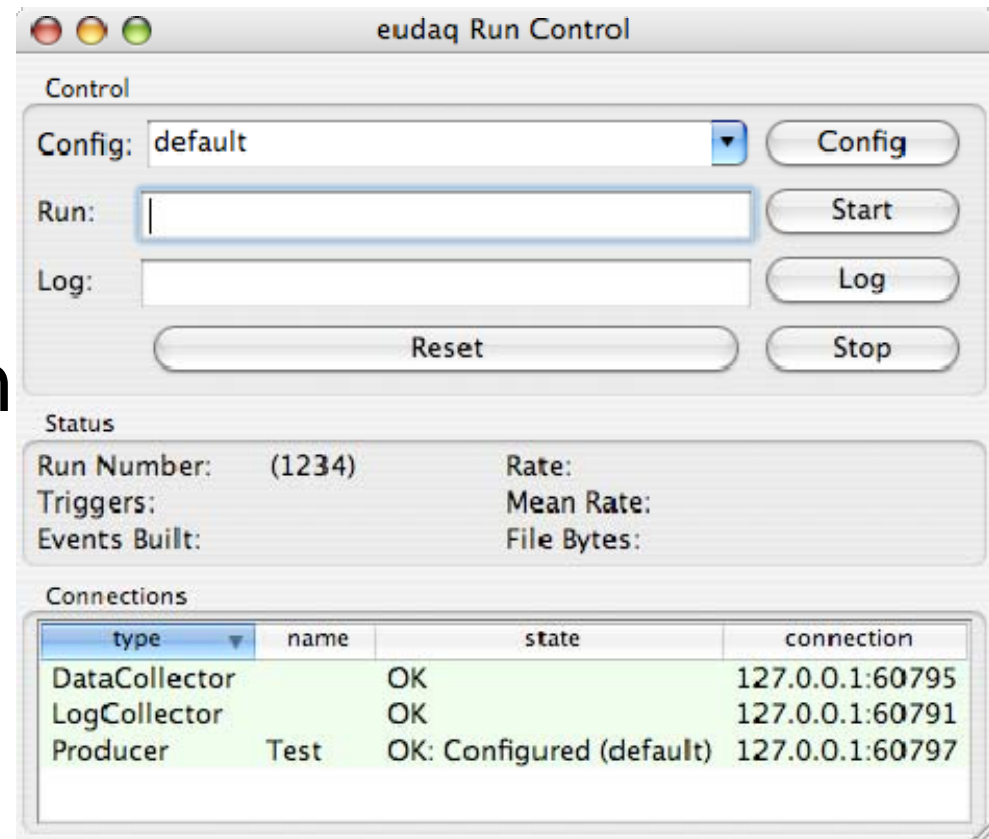
Data Collector

- Receives commands from Run Control
- Receives data from all the Producers
- Merges Events with same event number and writes them to file
- Currently writes custom binary format which is then converted to LCIO offline
- Should we write directly in LCIO?
 - How much processing? Clustering? Tracking?
 - Maybe do processing in a separate task
 - Or automatically copy data to DCache and start a conversion process on the Grid



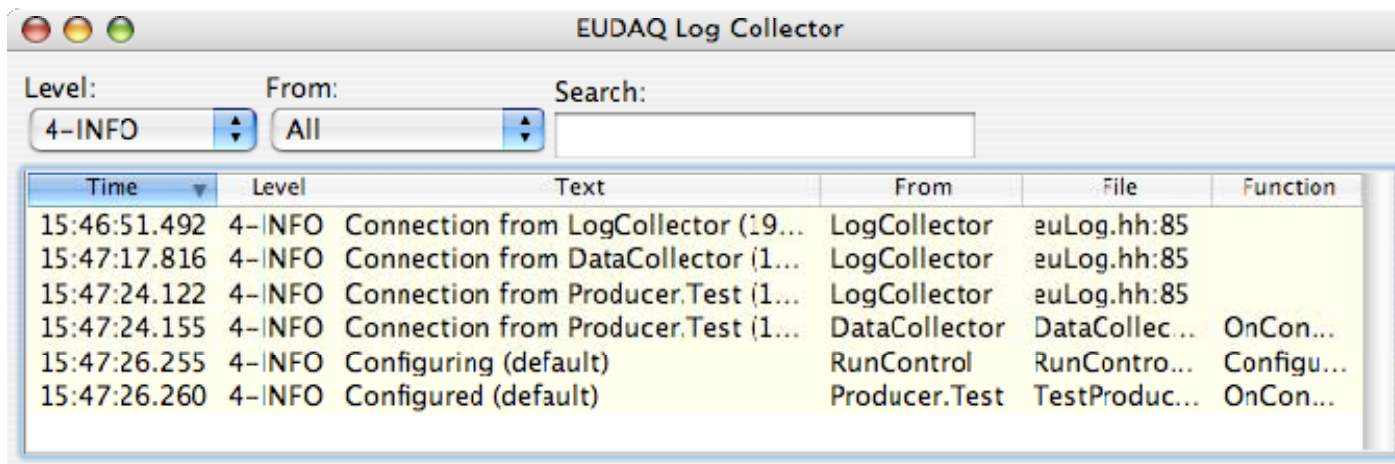
Run Control

- Controls the other processes
- Displays basic status information
- In progress:
 - GUI for configuration



Log Collector

- Receives log messages from all other processes, and saves them to file
- Keeps all log messages in a central location
- Can filter on level / source
- Can search in messages

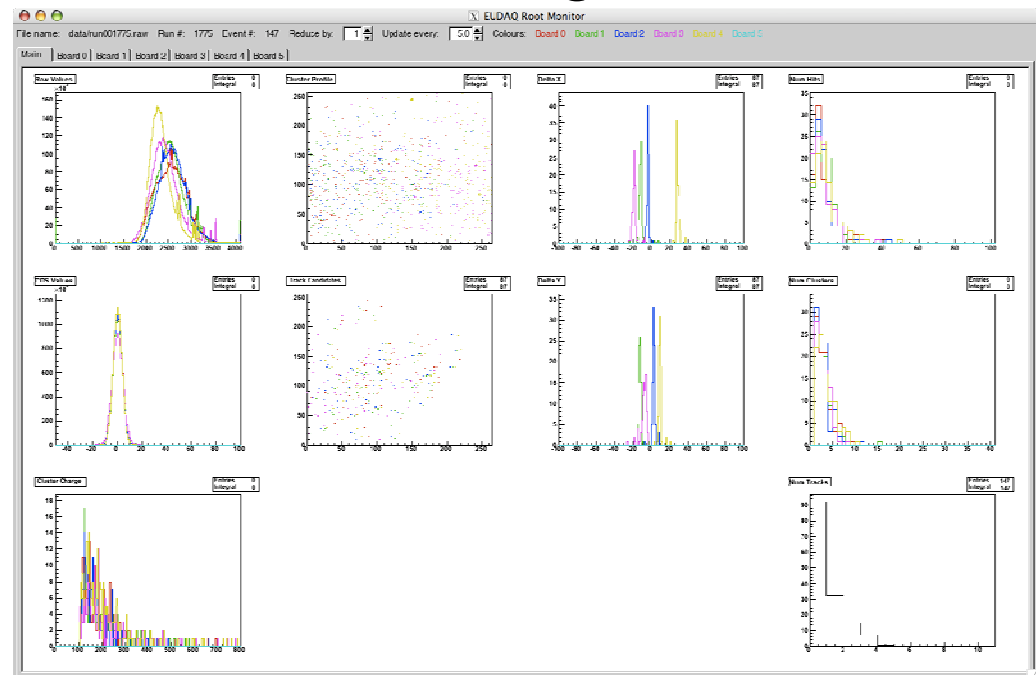


Time	Level	Text	From	File	Function
15:46:51.492	4-INFO	Connection from LogCollector (19...	LogCollector	euLog.hh:85	
15:47:17.816	4-INFO	Connection from DataCollector (1...	LogCollector	euLog.hh:85	
15:47:24.122	4-INFO	Connection from Producer.Test (1...	LogCollector	euLog.hh:85	
15:47:24.155	4-INFO	Connection from Producer.Test (1...	DataCollector	DataCollec...	OnCon...
15:47:26.255	4-INFO	Configuring (default)	RunControl	RunContro...	Configu...
15:47:26.260	4-INFO	Configured (default)	Producer.Test	TestProduc...	OnCon...



Root Monitor

- Reads data from the files on disk - doesn't slow down the DAQ if it gets behind
- Uses Root to generate various histograms
- GUI Completely Root-based (not very stable)
- Some settings still hard-coded



Remaining Issues

- Software repository: find a better location
- Speed: 3 Hz (RAW), 10-15 Hz (ZS)
 - VME Library: 2 ms per access (much too slow)
 - EUDRB: 1 event read out at a time
 - Producers: reading event and sending could be done in parallel
- Stability:
 - Threading: some resources are not properly protected with mutexes
 - Sometimes data taking stops (often after first event of run): needs more investigation



Summary

- Already have a reasonable usable DAQ system, but a few remaining issues
 - Test with 6 EUDRBs simultaneously
 - DAQ PC: Upgrade OS and RAID system
 - Software repository: move off DAQ PC
 - Improve speed and stability
 - Run Control: GUI for configuration
 - Data Collector: Processing - what/where?
 - Documentation

