Software for Slice Tests

Murrough Landon – 2 November 2000

Overview

- Status of DAQ -1
- “DetDAQ” plans
- Our requirements
- Integration plans
- Discussion items
- Effort
- November TDAQ Workshop

http://www.hep.ph.qmw.ac.uk/~landon/talks

DAQ -1 Status (1)

Online Software (aka BackEnd)

- Recent additions (version 0.0.10) include Monitoring Framework. User skeleton is provided (but missing a good Event class)
- Also some additions for the TileCal test beam.
- OKS Database now uses XML as internal file format. No changes apparent to the end user.
- Near future (by Xmas?): replacement of Rogue Wave Tools.h++ by STL. At that point it will be fully open source.
- Event Dump utility (using the Monitoring Framework) is under development in Java. We would still need to add subdetector specific decoding.
- New extended API for the Information Service (IS) is under consideration. Self describing data and integration with OKS. May not be completely backwards compatible?

DAQ -1 Status (2)

Dataflow Software

- Current: Tile Cal test beam used Readout Crate implemented by CES RIO2s: 3 ROBs, EBIF, LDAQ.
- Imminent: Readout Crate implemented on a PC. Slink/PCI cards available at 2.2kSF each. Typical PC has maximum of six PCI slots.
- This may provide a possible hardware architecture for slice tests (given money for Slink/PCI cards: would need to order soon from Erik van der Bij).
- Advantage: use complete DAQ -1 environment, automatic even building from multiple RODs without (?) large software effort.
- Medium term: joint work on “DetDAQ” (aka ROD crate DAQ) by LAr and TileCal, both now using the same RODs.
- This would involve developing a ROD crate controller: implement some dataflow functions (the “IO Module”) in software running in the crate CPU which would interface to the RODs.
- We should provide input into this activity.
- Eg DetDAQ sees no requirement for event building from multiple RODs.
- Outstanding issues: dataflow code is not yet open source. Some libraries have been released to us privately. More work needs to be done on documentation and packaging the software for non expert developers.

Slice Test Architecture

Hardware

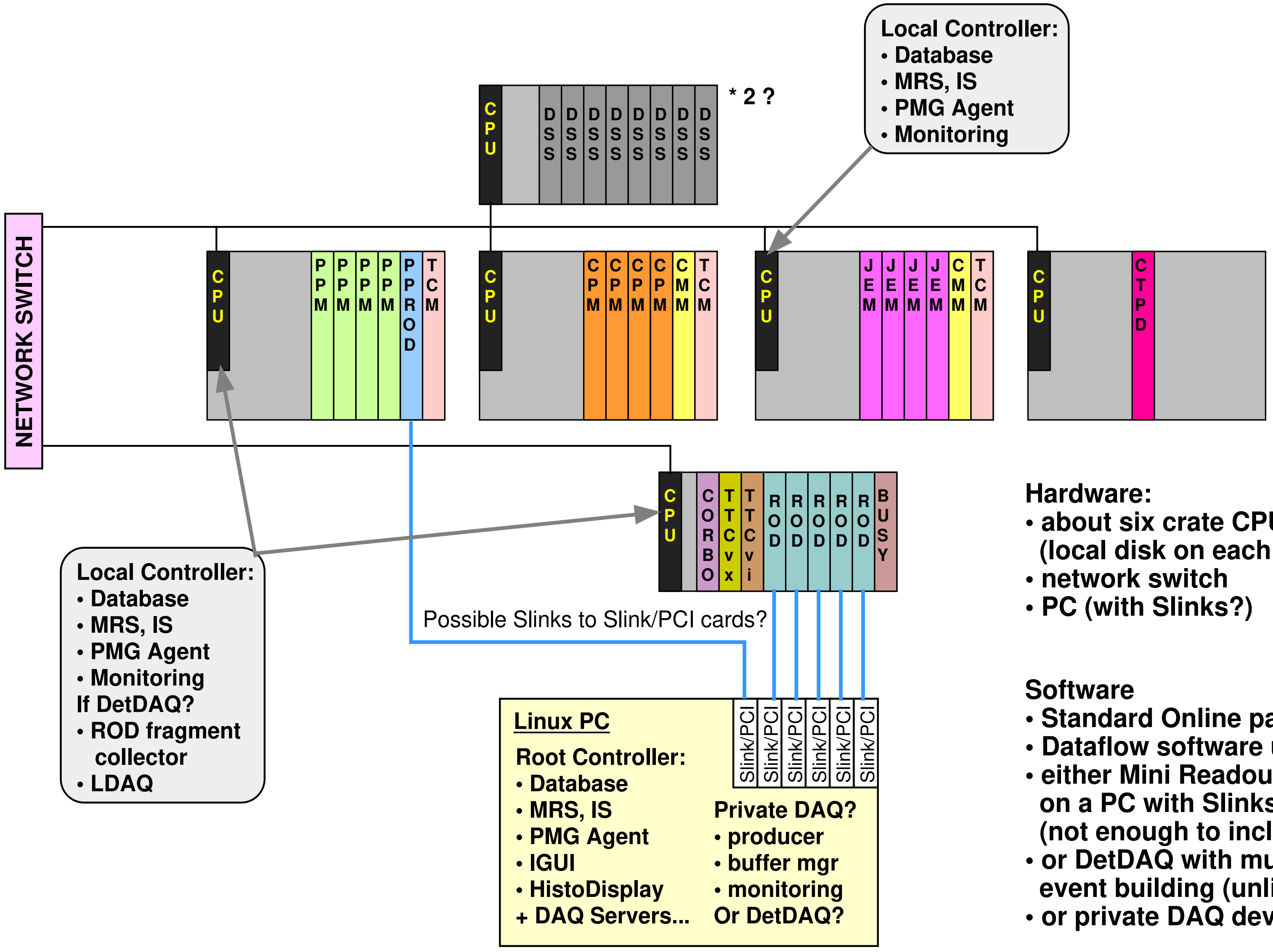
- Two or three 6U crates, two or three 9U crates (CPM and JEM in same crate?), possible CTPD crate
- CPU in each crate, no VICs!
- One or more PCs acting as control workstation
- Possibility of Slink/PCI cards in PC to act as DAQ -1 Read-out Crate

Software Packages

- Linux, usual system and CERN libraries, ROOT, etc
- DAQ -1 Online software
- DAQ -1 Dataflow libraries where available
- DetDAQ if ready
- HDMC (with continued development)
- Plus software we still have to design and write...

L1 Calo Trigger: Slice Test DAQ Setup

Murrough Landon
31 October 2000



Hardware:
- about six crate CPUs (local disk on each?)
- network switch
- PC (with Slinks?)

Software
- Standard Online packages
- Dataflow software uncertain:
- either Mini Readout Crate on a PC with Slinks (not enough to include CTP)
- or DetDAQ with multicrate event building (unlikely?)
- or private DAQ development

Required Software (1)

Run Control

- We need a Local Controller for each crate: provide code in the DAQ -1 run control skeleton to implement the actions for each state transition. Includes: checking hardware configuration, load calibrations and run parameters, setup calibrations...
- Single process (multiple threads?) running in crate CPU (but could run in PC and use net bus server).
- Read configuration (modules in crate etc) and calibration from database files and run parameters from IS.
- Load modules with appropriate settings for Physics/Calibration runs; load playback memories for some calibrations?
- Interface to DetDAQ in ROD crates?
- Continuous monitoring of local crate (transmission errors, rate histograms from PPMs, etc). Monitor data published in IS.
- Special overall run controller for whole system is probably also required (interface to trigger for calibrations?).

Required Software (2)

Readout

- Still not clear if DAQ -1 and (as yet non-existent Det-DAQ) will be sufficient for our needs or available on our timescale.
- Extend Bruces DAQ work *for ROD tests) to provide producer, buffer manager (based on PBM library) and monitor programs for multicrate slice tests.
- Producer could be single process running on a control PC with low level VME access via bus server; or try to move Module functionality into crate CPUs, eg with CORBA...?
- Need to decide whether to continue with private s/w development or try to join and influence the DetDAQ approach.

Distributed Histogramming

- Unless/until anything emerges from Online software group, continue to use modified CDF product developed by Tara.
- Future: possibility of ROOT histograms distributed via IS?

Required Software (3)

Monitoring Programs

- Event dump program: version in PBM environment in progress. Would require conversion to Java to be included in DAQ -1 event dump (if that is possible). Main effort so far is developing a smart Event class - not yet provided by DAQ -1.
- Analysis of events: aim to keep subdetector specific analysis separate from PBM framework, so it can be slotted into DAQ -1 framework later.
- Program to collect and display continuous monitoring data from multiple crates - ie subscribe to appropriate IS data.

Online Simulation

- Simulation of the detailed behaviour of the whole trigger. Also of partially complete slice.
- Overlap with offline simulation. Aim to develop that in such a way that parts of it can be used outside the ATHENA framework.

Offline Analysis

- Analysis programs (like previous test beam analysis programs) to read stored event data? Repackage online programs?

Required Software (4)

Calibration Programs

- Require setup code in the Run Controller.
- Monitoring programs to analyse the events and create new database files.
- Q: how to control the trigger? Special run controller?

Test Vectors

- Programs to generate test vectors for various checks:
- Readout of PPM/CPM/JEM/CMMs to RODs
- PPM to CPM/JEM connectivity
- CP and JEP crate backplane connectivity (CPM-CPM, JEM-JEM, CPM-CMM, etc).
- Ditto for timing setup
- So far we have imitation CPM data (to be loaded into DSS) to check ROD behaviour.

Required Software (5)

DCS

Not even considered yet...

Required Software (6)

Database Schema and DALs

- Still havent defined database schema...
- New DB classes for our Modules (or type field in standard Module?)
- Calibration parameters. Sensible organisation: one database file per crate for each type of calibration? Typically read and written by one process.
- Trigger menu information: thresholds, BCID algorithm choices, etc
- Run parameters into IS?
- We can always access DB information directly using the full DB schema. May be useful to define a Data Access Library (DAL) to provide convenient access to commonly required information. Simplified view of the database.

Database

Calibration Datafiles

- PPM crate: 16 * 64 channels of Energy calibration. May be slope and offset or full LUT contents.
- PPM crate: 16 * 64 channels of BCID parameters
- PPM crate: 16 * 64 channels of timing parameters
- CPM and JEM crates: channel delays?
- CPM and JEM crates: thresholds
- Also, common parameters: number of slices, readout pointers etc.

Configuration Data

- Crates and Modules: addressing info
- Connections: (phi,eta) space occupied by each module. NB PPM crates cover non-rectangular spaces.
- Alternatively, L1CaloMapper class to map connections by algorithm. Could have several for slice test setups?

Conditions Database

- Online calibrations will be stored in the offline “conditions database”. Mechanism for moving them there? (Not needed for slice tests though).

Calibration Procedures

PPM: Energy, BCID and Timing Calibration

May not be required/doable for the slice test? Requires Calo calibration systems - though analogue input of some kind will be available?

Can be done one PP/ROD crate at a time (but note orthogonal division of PP and Calo partitions).

PPM-CPM/JEM Timing and Connections

Needs coherent events read from PP RODs and CP/JEP RODs.

CP and JEP Backplane Checks

Can be done one ROD crate at a time.

CMM-CTP Transmission Checks

Needs events from at least two ROD crates.

Trigger Functionality

Needs coherent events from all available ROD crates.

Performance? For test data, rate will be limited by the speed of loading new test vectors.

Load 128 slices in all PPM channels. Run 8 L1As then stop triggers. CTP deadtime then gives 88us to load next 128 slices in 64 channels in 16 PPMs per crate (about 128kb/crate).

Q: for final system, if we could run at max L1 rate, can we use custom level 2 algorithm to check expected outputs?

Calibration via RODs or Modules

Using modules and freezing pipelines

Simple idea: fill playback memories: start synchronously, then stop and freeze pipelines synchronously (via TTC command) after 128 clocks. Read whole pipeline from each module.

Could perform eg timing setup, connectivity check with one event.

But requires software development just for reading and analysing this data.

Using RODs

Uses standard DAQ software: we will have to be able to read the ROD data anyway.

But can only read five slices. Need several events to read the whole playback memory (which is useful for checking connections with special test vectors).

Workaround: send L1A every 512+5 ticks (max L1A rate). Cover whole pipeline in 26 events.

Other Activities

Hardware Infrastructure

Gain experience with new CPUs (Mainz now, QMW soon, others?). About five CPUs required for slice tests?

Dont forget documentation!

Integration Plans

DAQ -1 and Level 2

Plans are well advanced to integrate level 2 into DAQ/EF -1 at both hardware and software level by summer 2001.

Level 1

We are invited to draw up our own plans for integration....
From the adhoc report:

```
Phase 1

1. The LVL1 calo ROD and muon ROD (MIROD) must be integrated
   with the LVL2 Region-of-Interest Builder. This integration
   will be performed first in standalone tests in late 2000
   or early 2001.

2. At least one LVL1 ROD should be integrated with a ROB in
   standalone tests. This integration will be performed first
   in standalone tests in late 2000 or early 2001.

Phase 2

1. Vertical slices of LVL1 calo and LVL1 muon trigger systems,
   including front-end electronics, should be integrated. This
   is an internal LVL1 integration that will occur during 2001.

2. Integration of one or more LVL1 RODs, and of ROD_BUSY and
   TTC modules, in the HLT/DAQ integrated prototype.

3. Integration of at least one vertical slice of LVL1 into
   the complete integrated prototype system, initially in the
   lab and then at a test beam using real detector signals.
   This integration should be performed in early 2002.
```

Discussion Issues

H/W and S/W Architecture

- Should we use PC Readout crate? Or software solution?
- Continue with private developments or move entirely to DetDAQ?
- Both the above may turn on collection of same event from multiple RODs.
- Interaction of single process PBM Producer with distributed Local Controller in each crate?

Priorities

- Where should we direct our limited effort?
- General strategy: use DAQ -1 code and APIs wherever possible even if we have to continue our own private developments.

Effort

TileCal Experience

Used pure DAQ -1 solution (ie not private DAQ or DetDAQ). Took 4-5 full time people about 4-5 months to develop their run controllers, ROD software, monitoring and offline software.

...plus considerable support from several members of the DAQ group.

L1 Calo People

UK: Bill, Bruce, Murrough. Soon rejoined by Stephen Hillier; Scott still leaving...

HD: Cornelius (ASIC permitting?)

November TDAQ Workshop

Presentation

- Two hour (parallel) session on Tuesday 14th
- Discuss level 1 requirements and use of Online software
- NB DetDAQ has separate session on Wed 15th

Preparation

- Document to gather information...
- Use cases describing our requirements (short & long term)
- How are these achieved at the moment
- Assessment of the Online packages
- Future programme of work

Feedback and questions

- Generally positive appreciation of Online packages and documentation.
- Who will be responsible for subdetector database schemas, DALs, GUIs?
- Any plans for distributed histogramming?
- Subdetector extensibility of eg Event Dump
- Ditto sophisticated Event class?
- Refinement of IS API
- Online packages APIs also available from Java
- Interaction of Run Control with calibration triggers?
- Continuous monitoring vs RC partitions?