Prototype DAQ Software

Murrough Landon – 11 November 1999

General Aim

- Develop a prototype of the final DAQ system
- ...including setup, monitoring and calibration
- Provide a DAQ system for tests of prototype ROD
- Platform for joint tests with level2, DAQ
- Gain experience with ATLAS DAQ software

Requirements

- Read events from our RODs and other modules
- ...assuming a hardware mechanism for capturing them
- ...eventually from multiple crates (via network)
- Provide facilities for monitoring events
- Use DAQ -1 components where possible
- ...follow (evolving?) DAQ -1 designs otherwise

Available Components

Available Components

- The “Back End” DAQ -1 components have been released (CDROMs) for general use.
- But, despite many requests, DAQ -1 “Dataflow” implementation is not (yet) suitable for use in a small scale setup, eg a ROD crate.
- So, we will have to provide the dataflow side ourselves for the moment. We could reuse our old (C) buffer manager, with new producer and monitoring applications (in C++).
- These can make use of Back End components such as: database, message reporting, information service, run control, process manager.
- For histogramming, we can use ROOT. This offers facilities for distributed booking, filling and display of histogram (and other?) data.
- For code specific to our modules and for VME access etc, we can base developments on the HDMC software.
- Control TTCvi through newly developed classes.

DAQ -1 Back End Software

Experience

- Last december: brief trial of the OKS database tools.
- OKS editors are somewhat cumbersome to use, but recent developments include a new “partition editor” using Data Access Library.
- No work yet done on defining L1Calo database - this will be a significant task.
- Recently tried most other core components from CDROM and via AFS.
- Message Reporting System (MRS): very easy to send error messages. Also simple to receive them (dedicated error message server is also provided). Didnt try filtering messages.
- Information Server (IS): simple dynamic database (dictionary) for run time exchange of data between processes. Easy to publish and query information. Process must implement a CORBA server to receive notification of updates. NB the IS data is persistent.
- Run Control (RC): a more substantial component. All tutorial examples tried (mostly successfully - after a little help). Noddy run controller built from the skeleton.
- Process Manager (PMG): not yet tried - but looks OK...
- Integrated GUI (IGUI): works - but still needs a startup script. To be useful to us, it probably needs the capability for us to customise it. Plan is to provide a Java Bean API - but this is not yet implemented.

DAQ -1 Back End Software

Evaluation

- In some ways its clearly still a prototype
- Its also still in early versions as an integrated package (some rough edges)
- But the documentation is usually comprehensive with many examples
- In some cases more general introductions and explanations would help
- Attempts to future proof it mean it consists of several layers which increases the complexity and amount to learn
- OKS database tools need considerable improvement to be used on a large scale. The new partition editor has not yet been released
- The Integrated GUI is still under development
- Conclusion: it **is** in a state we can use, but still needs more work to be pleasant to use

Feedback

- I have sent many long mails with comments on documentation and the software itself – most receive long and considered answers
- Several minor bugs and problems reported – fixes (or promises) received fairly promptly
- Considerable help from Bob Jones to find a typo in my modifications of the configuration database
- Overall impression is quite positive

Interactions with DAQ Group

DIG Meetings

- Norman (and I sometimes) have attended several of these meetings.
- Intended for exchange of information, views, requirements between the subdetectors and the DAQ group.
- We have presented our requirements for DAQ in the ROD crate. The LAr community has also been pressing for this (and offered up to five people to help develop it). So far little response.
- Apart from this disappointment, its an important and useful group.
- Next meeting at Beatenberg.

Other Meetings

- Private discussions with members of the Dataflow group, eg Dave Francis.
- Positive atmosphere, but it is clear that their priorities are elsewhere.

Future Workplan

Future Workplan

Soon Refine requirements: what do we need for the ROD?

Soon Define database classes specific to level 1 (in conjunction with Muons, CTP) and those specific to the calo trigger

Later Develop GUI editor to fill our database (data access library?)

Soon Consider run control states, define producer and monitor apps

Soon Implement producer and monitor applications

Soon DSS, TTCvi, ROD in HDMC framework: as Modules?!