Run Control, ROD Crate DAQ, Run Types, Processes

Murrough Landon – 2 March 2005

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

Overview

- L1Calo run control status
- ROD Crate DAQ and recent developments
- Run types?
- Starting standalone processes
- Proposals

L1Calo run control status (1)

What we do at present

- Customised run control skeleton: access to main and state transitions
- Initialisation and actions common to whole crate: eg read database and IS once
- Transition actions propogated to modules (as RCD)
- Starting/stopping processes: at the moment only as specified on command line. Envisaged to have run type dependent control, but never implemented
- Direct access to IS for reading parameters and storing module status
- No readout or event fragments from RODs or data from non-ROD modules
- No filling and publishing of histograms (by run controller)
- Custom controllers also for simulation and L1Calo root (starts sequencer)

ROD Crate DAQ Status (1)

Overview of the RCD model

- Based on the ROS software: handle triggers, readout ROBINs, monitor and transport fragments
- Implemented as a set of plugins configured by database – no access to main
- LocalController skeleton controlling separate IOManager application
- Database read by the framework, configuration parameters per module passed in maps – allows IOM to be run independent of Online
- **New!** Also have direct access to configuration database
- Transition actions (and asynchronous commands?) propogated individually to RCDModule plugins – need a workaround for common crate actions

ROD Crate DAQ Status (2)

Overview of the RCD model (continued)

- **New!** Transparent access to IS and OH APIs: two methods called automatically at configurable frequencies to publish eg statistics and histograms
- Facilities for reporting errors including catching and handling error conditions
- Readout of “DataChannels”, eg event fragments from RODs (or ROBins) or in principle from other modules(?)
- **New!** Separation of readout for dataflow and for monitoring including better handling of sequential fragments eg from RODs
- **New!** Support for interrupt handling from RODs (do we need this?)

Run types

What are our requirements?

- Presently DB implementation a bit complex (see Database talk)
- We hardly use separate module options, nor run type specific TTC commands
- Dont yet use run types to define multiphase (or multistep) runs
- We do use them to set playback modes and feed test vectors to different module types (DataGenRecipes)
- Joint calibrations with calorimeters will need something similar
- Can we define what we want in future?

Starting processes (1)

Not via RCD?

- No nice place to add starting of our own processes in RCD
- Could do it but would be a bodge
- Better to move to standard model – start processes directly via online framework (currently DSASupervisor)?
- At present no facility for starting processes or not depending on an IS parameter, eg IGUI run type choice

Starting processes (2)

Possibilities?

- Some recent email exchanges with Giovanna Lehmann – she is willing to add something simple if we can decide what we want
- My ideal: application attribute “startIf” some boolean IS and/or database parameter is true
- More cumbersome, some IGUI code uses IS choice to update sets of enable/disable flags on database objects (and stores database for reloading by DSASupervisor and run controllers)
- More baroque: applications controlled by database Resource objects
- More untidy: start all processes and let them decide if they want to do anything given the run type – cant let them die as online would try to restart them or declare an error

Proposals and questions (1)

Move to RCD

- We have delayed this for a long time: how to start “kicker” processes is one of the reasons, also not wanting to change in mid-slice test, some worries about IS access and general lack of time to try it more
- Should we use RCD readout of non-ROD modules for what are now implemented as kicker programs?
- RCD readout of RODs for ROD fragment monitoring certainly useful but not a priority since we have ROS based monitoring?