L1Calo View of ROD Crate DAQ

Murrough Landon – 17 November 2004

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

Contents

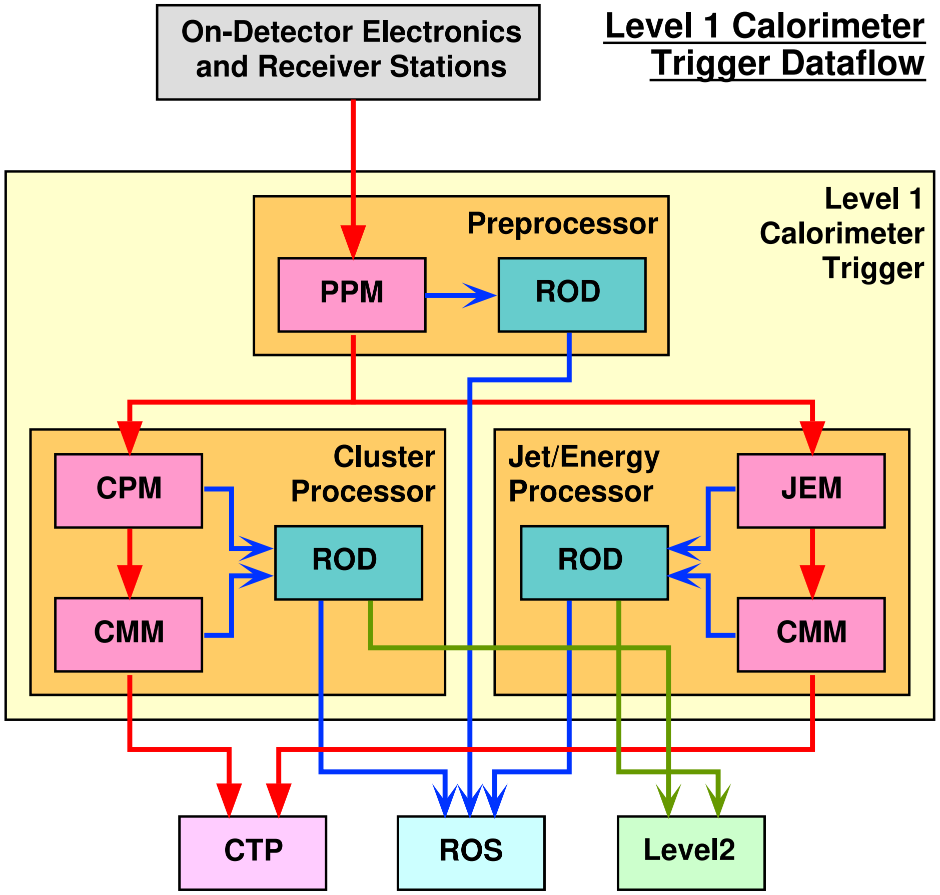
- L1Calo Introduction
- Existing L1Calo Software
- Expected Usage of RCD
- Comments on RCD
- Future Plans

On behalf of the L1Calo group

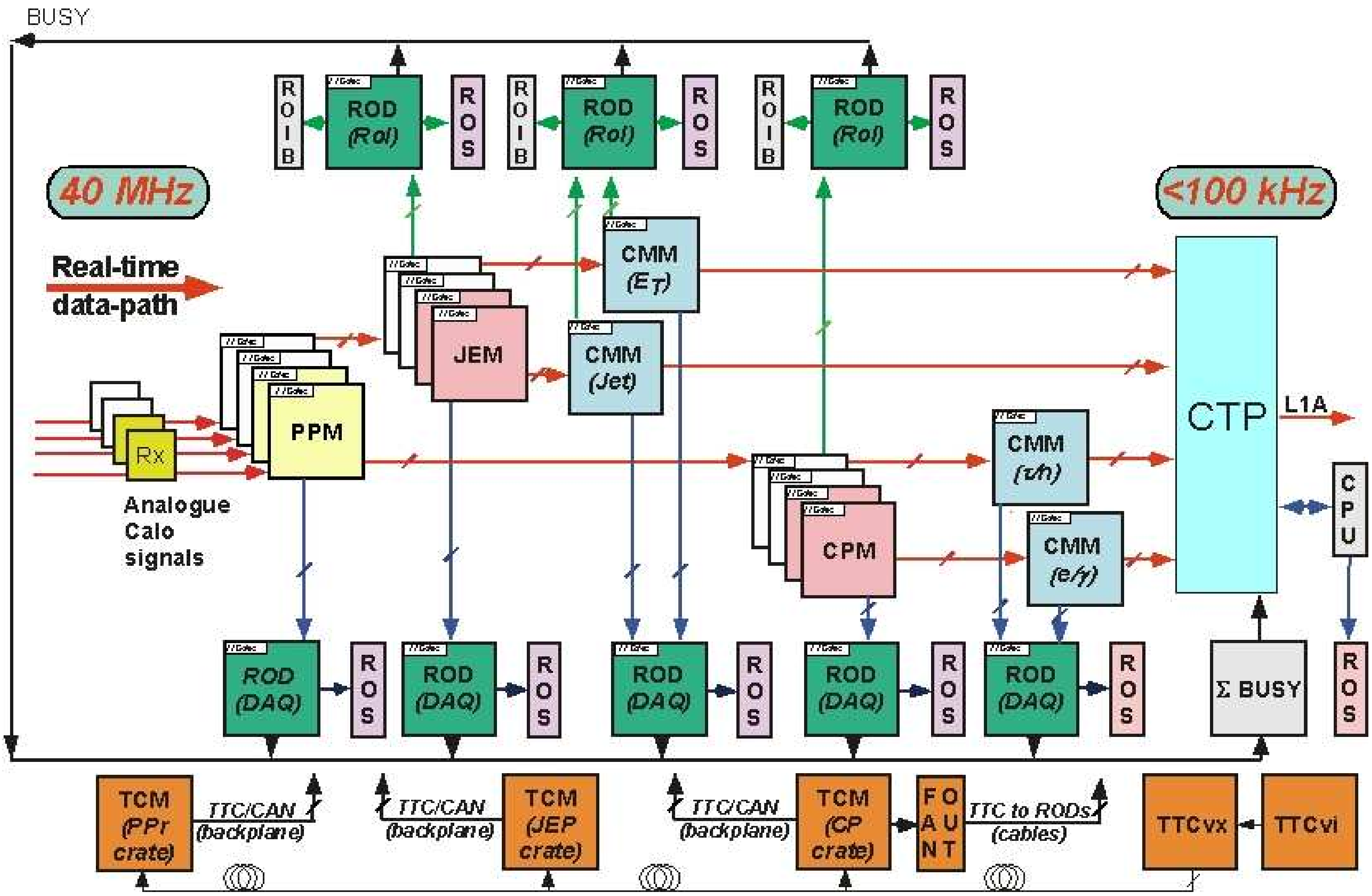
L1Calo Introduction

Overview

- Three main subcomponents: preprocessor, cluster processor and jet/energy processor – plus RODs, TTC infrastructure, cables, etc
- Real time data to CTP. RoIs to LVL2 and readout to DAQ via RODs
- About 15–20 crates, some with “reduced VME” backplanes
- Six types of custom 9U module with variety of firmware personalities plus test modules



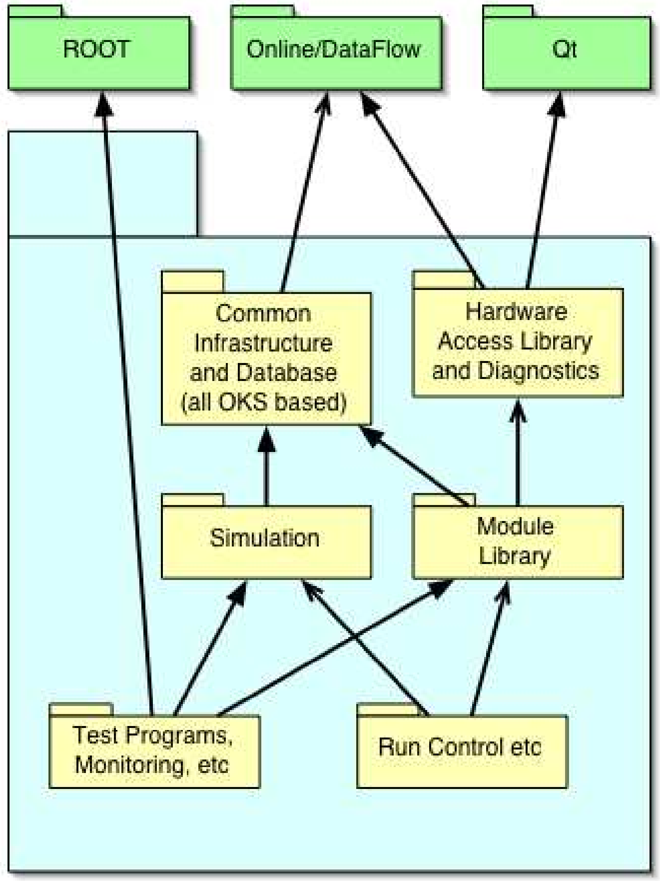
L1Calo Testbeam Setup



L1Calo Software (1)

Overview

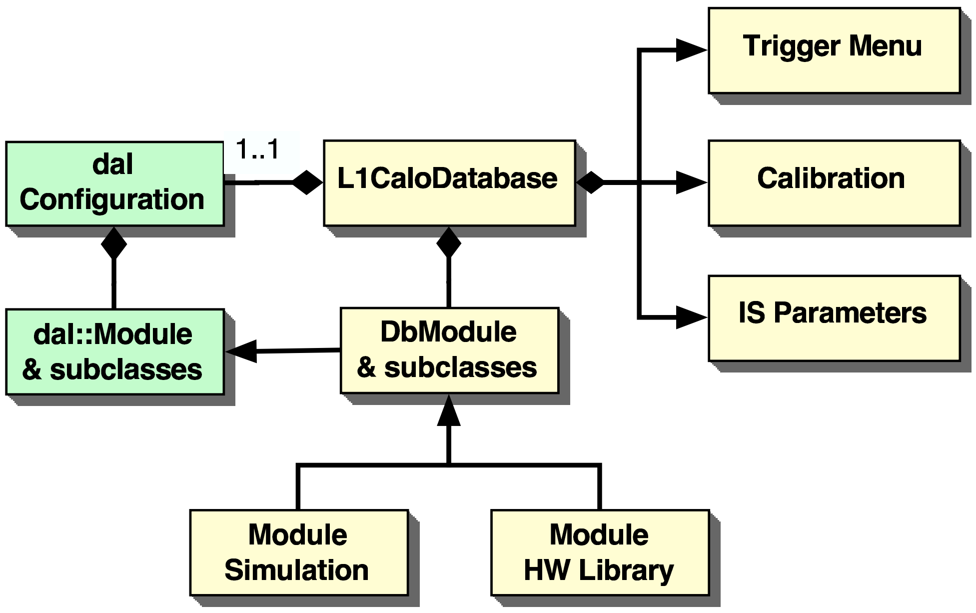
- Developed for slice test programme, but with final ATLAS in mind
- About 50 CMT packages in common TDAQ CVS repository. Nightly builds, Doxygen, etc



L1Calo Software (2)

Databases

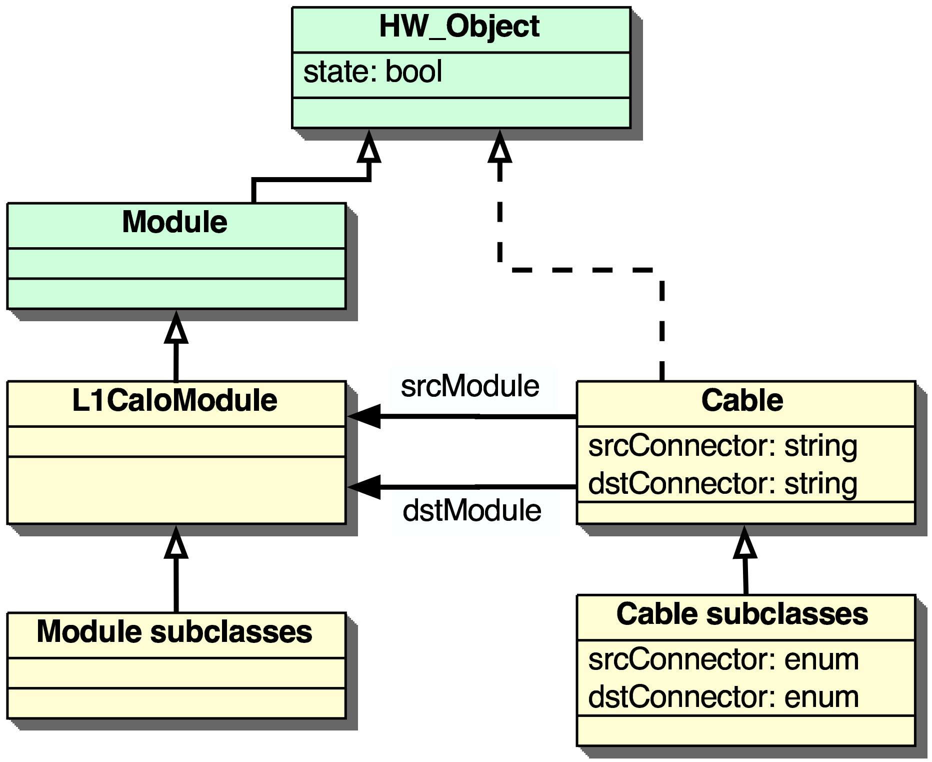
- Many extensions to OKS configuration database
- Module subclasses, cables, firmware, runtypes, test vector configuration
- Temporary OKS implementations of trigger menu and calibration (pending clear ATLAS DB strategy)
- L1Calo DB: additional layer to shield module/simulation code from (past and) future changes



L1Calo Software (3)

Cables

- L1Calo has many internal cables: description of connectivity is vital for simulation
- Also required for hardware setup (mask inputs/outputs if cable or remote module is disabled)
- Common, easily maintainable schema for cabling is needed
- Eg disabling a ROD should automatically disable BUSY input and ROS input channels



RCD Status

Lab tests

- Our long running “slice tests” are inherently multicrate
- Online SW used for many years and the ROS for a year or so
- Tried migrating run controllers to RCD style in the summer
- Gave up chasing obscure segmentation fault in ROS/RCD despite help

H8 implementation

- Used existing run controllers (updated for new state model)
- Readout with four prototype RODs via Slink (FILAR)
- Monitoring of events via the ROS

Expected RCD Usage (1)

Configuration and local monitoring

- Main use: configure trigger processor modules and their readout
- Perform local hardware and operational monitoring (rates, link errors, onboard histograms)
- Probably sample events from the RODs especially for calibration, but most useful event monitoring is at ROS or SFI level

Plugins

- Single(?) custom plugin for generic L1Calo module configuration with link to existing hardware configuration
- Standard dataflow plugins for monitoring, trigger, readout

Expected RCD Usage (2)

IS parameters

- Few (1–4) parameters per module (eg enable/disable) to override ConfDB (implemented before IGUI panels could write to the database)
- Some choices: trigger menu, calibration, L1Calo run type, test vector generation
- Operation statistics per module and from monitoring/checking programs
- IGUI panels are constructed automatically using IS metadata. Adding an attribute only requires editing the schema and setting/reading the new attribute in the module library. Any architecture which requires repackaging of IS information for transfer between LocalController and IOManager would entail extra maintenance

Expected RCD Usage (3)

OH

- Presently only used by monitoring programs running on workstations
- In future crate CPUs will read and publish histograms created by ASICs and FPGAs

DCS

- Useful for us (just monitoring, not configuration) but not a priority

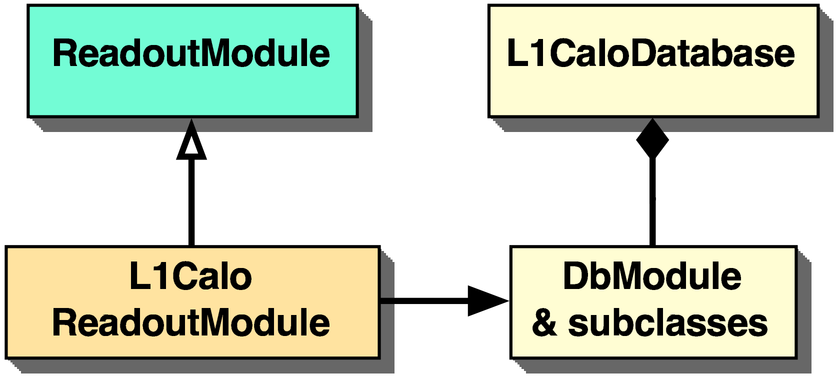
Calibration

- Envisage multistep runs, synchronisation between crates
- Joint calibration with calorimeters (more discussion needed!)

Expected RCD Usage (4)

Databases

- Passing configuration from local controller to IOManager via parameter maps is not helpful for us
- Our database is an object structure: we will (re)read it in IOManager
- In future most of our data will come from conditions DB, not online configuration DB
- If the present confusion of multiple databases converges on O(1) solution, a standard RCD access to conditions DB would be useful



RCD Comments (1)

Databases

- ROS/RCD requires proliferation of DB objects
- Eg adding a module in existing L1Calo software needs two steps: create new Module object and link to Crate object. Run controller configures all Modules in the Crate
- With RCD we need the above (for hw and cabling information) plus: create new ReadoutModule and link to RCD object and hardware Module object.
- Some merging of online DB hardware schema and RCD schema would be good
- DB is read at initialisation, so you need to shutdown after modifying the DB (original Online SW advice was to load DB at Load step)

RCD Comments (2)

Useful but missing functionality

- In IOManager a common action before and/or after the ReadoutModule actions for each state transition
- A standard configurable mechanism for driving the run control
- Eg for calibration runs with many steps needing intermediate commands that must occur synchronously across crates
- Typical commands are to TTCvi: send a particular broadcast for this type of calibration/test procedure before/after this transition
- Run control “user” commands should be propogated to ReadoutModules
- ConfDB editor improvements: eg create pattern of several objects, duplicate and intelligently rename a collection of objects (eg crate)

RCD Comments (3)

Documentation, support, bugs

- General comment: documentation is not nearly as comprehensive as for Online SW
- There needs to be more detail on what various plugin options do and how to configure them in the DB
- Even recent documentation seems wrong(?): my segmentation fault was finally solved by following a real life testbeam example and using the “wrong” TriggerIn plugin according to user guide recommendations
- LocalController/IOManager split and multithreaded nature makes debugging hard
- Bug reporting mechanism for Dataflow/RCD (Savannah?) has been promised for ages but still doesnt exist (as far as I know). Bugzilla for Online SW has been useful in providing feedback

Future Plans

Software development

- Priority: finish prototype testing with existing software
- Meanwhile check more of the RCD functionality (readout and monitoring)
- Someday migrate databases when DB strategy is clear and tools exist

Installation and Commissioning

- Recent proposals to be discussed at L1Calo meeting next week
- Initial reaction: we would greatly prefer Slink/ROS readout rather than VME/RCD readout (though our ROD is capable of it)
- Especially for integration work with calorimeters in 2005 (needs more discussion with calorimeter groups too)