Software Status

Murrough Landon – 6 July 2002

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

Overview

- Overall Aims
- Packages
- Overall Test Organisation
- Integrations
- Software Organisation
- Evolutionary Delivery

The Grand Plan

Aims and Intentions

- The software used to test the prototype hardware should itself be a prototype of the final software.
- So we use the Online software, ROS, and other emerging ATLAS software tools.

How to perform a test

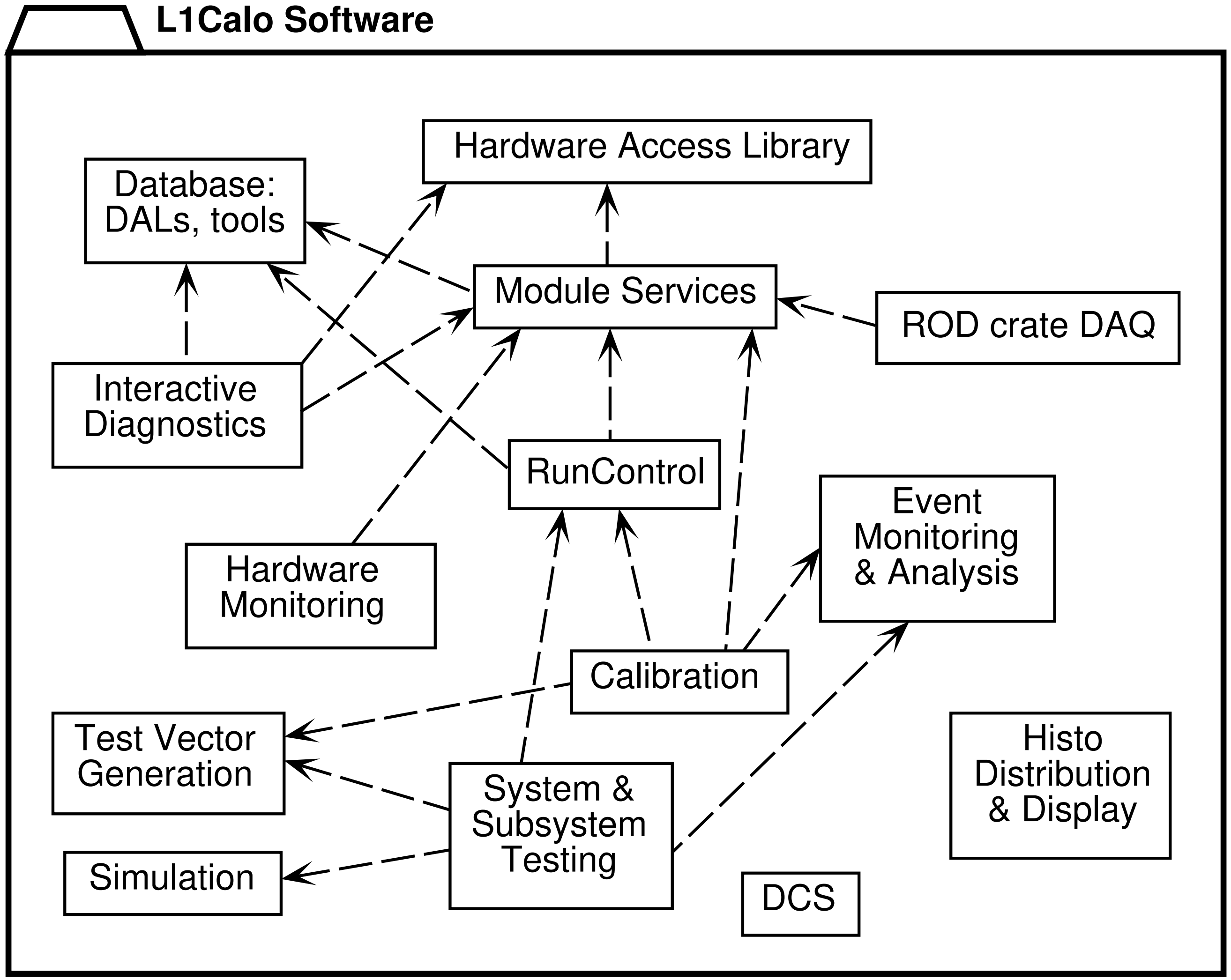
- Configure distributed multicrate system from databases under the control of the ATLAS run control system.
- Generate and load test vectors into parts of the system.
- Run, read out data and compare actual outputs with that from a simulation of the system under test.
- Use interactive diagnostics to peek at the running system.

Review of the Last Year

- As the hardware timescales changed, ideas for the software have evolved.
- Last summer we rewrote and reviewed our original requirements document.
- We also classified the work we needed to do into a number of clearly defined software packages.
- More recently we have agreed the overall scheme for organising tests and sets of tests.

Packages Diagram

L1Calo software packages and internal dependencies



Packages (1)

HDMC (Hardware Access and Interactive Diagnostics)

- Most of the code handling the extended parts file syntax (to support "composites") has been moved into the module services layer.
- This provides a hierarchical template style description of the substructure of a module, submodules, etc.
- Parts files and register definitions provided for more modules.
- Some new parts have been (or are being) developed recently: loading memories from files, loading flash RAMs, interface to TTCrx.
- New bus part using the CERN VME driver for Concurrent CPUs.
- HDMC CVS repository moved to Rutherford after Heidelberg firewall was tightened.
- Other longer term improvements to HDMC are still needed.
- Concern about future maintenance of HDMC.

Packages (2)

Module Services

- “Module Services” provides a higher level view of the functions of each module and its submodules for use by the DAQ.
- Implemented as a core package plus small packages for each type of module.
- Module services for DSS and CPROD are now fairly mature. Those for the CMM and CPM are being developed. Simple packages for TCM, TTCvi, etc also exist.
- Draft documentation (requirements) and examples exist.
- Module services packages recently moved into CMT.

Packages (3)

Databases

- Standard online configuration database extended to include classes for our modules and to describe the connections between them (single cables or groups).
- Information Service (IS) classes have been defined to describe L1Calo specific run parameters.
- Trigger menu (without editor) has existed for some time, but integration with CTP led developments will be needed.
- A preliminary calibration database also exists, but needs further development.
- Recently an “integrated” layer has been added to combine all the above data for simple use by module services objects. The aim will be to hide any future changes (eg move to the “conditions database” and use of the new CTP trigger menu developments) from the rest of our software.
- The API for this is reasonably well documented via Doxygen.

Packages (4)

Run Control

- Run control package has been fairly stable for some time.
- Interface to the database has evolved with its changes.
- Mechanism to start subprocesses has been added. Immediate use is for the CPROD tests. In future this will be useful to start a sequencer program to control multi step calibration or test runs.
- More work is required on the sequencer program.

Run States

- Software note 001 had suggestions for what we need to do at each run state transition.
- In recent discussions it is clear this needs to be updated.

Packages (5)

Integrated GUI and Information Service

- Also little change recently.
- The main run control GUI (the Online IGUI package) was extended a while ago to include L1Calo specific panels.
- These can set and display run parameters for L1Calo tests and for individual modules.
- Displays of module status (errors, etc) are under development.
- Classes using the Online IS package have been defined to store our dynamic configuration data, eg L1Calo specific run parameters, module parameters that it might be useful to change quickly (without editing the database). Module status data is also stored in IS.

Packages (6)

Simulation

- Core simulation package and module specific packages for the CPM and CPROD have been stable for some time.
- These packages have recently been moved into CMT.
- Work on CMM and JEM simulation is also in progress.
- Extension to the PPM and PPROD is required. Also extension of the CPROD to read CMM and JEM data.
- Current work includes implementing the proposed L1A scheme to control automated tests and a new interface and organisation for test vector files.
- The simulation is also being integrated with the updated configuration database.

Packages (7)

Test Vectors

- Code to generate test vectors for the CPM and CPROD has been around for some time.
- Work on generating test vectors for CMM has just started.
- Pulse library for PP system exists. Detailed test vectors for PPrAsic and MCM tests, but not yet for a whole PPM?
- Some test vectors exist for the JEM.
- All the above are still for single modules or small systems.
- Software to generate test vectors for larger diverse systems is still required. Eg for test data for four PPMs feeding two CPMs plus data for two flanking CPMs providing the environment, etc.

Packages (8)

Missing Packages

- Little or no work yet on event monitoring.
- Nor hardware monitoring.
- Discussions on calibration procedures with LAr and TileCal just started.

Untried Online Software

- Event dump has been briefly tested.
- Event monitoring and histogram (transport) package tried briefly at the DIG training.
- Online Bookkeeper not tried (could be useful for recording what tests we have done under what conditions).

Overall Test Organisation

Detailed scheme for running tests

- A meeting earlier this year came up with proposals:
  `http://www.hep.ph.qmul.ac.uk/l1calo/sweb/meetings/2002/testvectors.html`
- Use the database to define sets of named tests which will use selected calibration data, trigger menu, run parameters and test vectors on a particular hardware configuration. The simulation and the run control need the same data as input.
- Work on the database aspects has started.
- An interface has been defined for both simulation and run control (module services) to access test vectors.
- More integration work is still required...

Integrations (1)

Whats been done

- Integration of module services with run control (twice with different versions)
- Integration of module services with the database (not quite complete)
- Integration of simulation with the database (twice with different versions). To complete this properly needs the interface to test vectors to be implemented.
- Hopefully the latest round has been with more stable versions than the first times (where valuable feedback was obtained)
- Recent efforts with module services mean that the standalone CPROD tests ("Looper") can now be run via the run control with the configuration taken (mostly) from the database. And it almost works...
- Initial integration with the ROS.

Integrations (2)

Whats remains to be done

- Integration of simulation with run control: run the (test vector generation and) simulation for the chosen configuration to produce the test vector input and output files needed by module services.
- Integration of simulation outputs with module services (ie via agreed interface for reading test vectors)
- Use of the ROS for event monitoring.
- Implementation of event monitoring tasks to check the data against simulation outputs.

Software Organisation (1)

CMT

- We now have a total of 22 packages (some very small) using CMT as the build tool. CMT is very useful in organising collections of many packages developed by different people.
- The working model is basically copied from the Online group with some extra scripts from the DataCollection group and some developed ourselves.
- Trigger/DAQ as a whole aims to develop a single working model for using CMT.

CVS Repository

- Now HDMC and everything else is all at RAL.
- But we currently have some problems with external access to the RAL repository via the pserver.
- If these cant be resolved soon, moving everything to CERN sooner rather than later may be best...

Software Organisation (2)

Website

- The software website aims to provide information to developers. A lot of feedback has been received recently. More information and tips on how we use tools like CVS and CMT is needed.

Meetings

- Monthly video conferences. Not very good for brainstorming, but better than nothing. So far only two ends (which is easier to arrange). But in practice this has been either Mainz or Heidelberg. UK people have all travelled either to RAL or Birmingham. We have never included Stockholm.

Software Training

Visits and Documentation

- People developing module specific code based on the core packages need to know how to do this.
- Developers must provide documentation.
  - Reference documentation generated from the code by Doxygen is (becoming) available for most packages.
  - But user guides are still lacking (apart from the simulation).
- Visits for joint working sessions will be useful (eg Thomas recently visited RAL).
- In the past we organised a special training session. (It may be difficult to do this at the right time for everyone).

System Management

Can we standardise?

- Suitably configured systems are important for productivity.
- CERN has standardised on RedHat Linux and will move to version 7.2.
- ATLAS software will only be tested on RedHat. Other Linux distributions may have subtle differences which require workarounds that have to be reapplied to every new release. Following CERN is likely to be easiest in the long run.
- Crate CPUs and desktop machines need a complete installation (all development tools) and should share a common file system. AFS clients would also be useful.
- Bruce has tested a scheme for network booting of crate CPUs.
- We have scripts to ease the installation of CMT and the Online software.

Evolutionary Delivery (1)

Overview

- Try to plan what aspects of the software will be available when.
- General plan: develop the CP system first.
- Extend to the JEP as its fairly similar and JEM0 is available. (Requires new firmware for CPROD and CMM).
- Later incorporate the PP system.
- Extension to JEP and PP systems can start before all stages in development of the CP system software are complete.
- The desire to get the software working should not be allowed to prevent necessary documentation being written!

Evolutionary Delivery (2)

CP System

- Complete integration of CPROD tests: module services, database, run control, test vectors interface.
- Be able to successfully run a single test via the run control. [July?].
- Add CMM and/or CPM with their test vector generators and simulation. Include L1A generation via a DSS.
- Be able to run a single test including simulation of the selected configuration. Event readout still via DSS? [August?].
- Integrate the ROS, implement event monitoring and comparison via the ROS.
- Expand to whole CP system. [September?].
- Implement sets of tests and test sequences.
- Develop timing calibration procedures (CPM and CMM inputs).
- Add hardware monitoring and reporting via the IGUI.