DAQ/Online for MAPS

Paul Dauncey

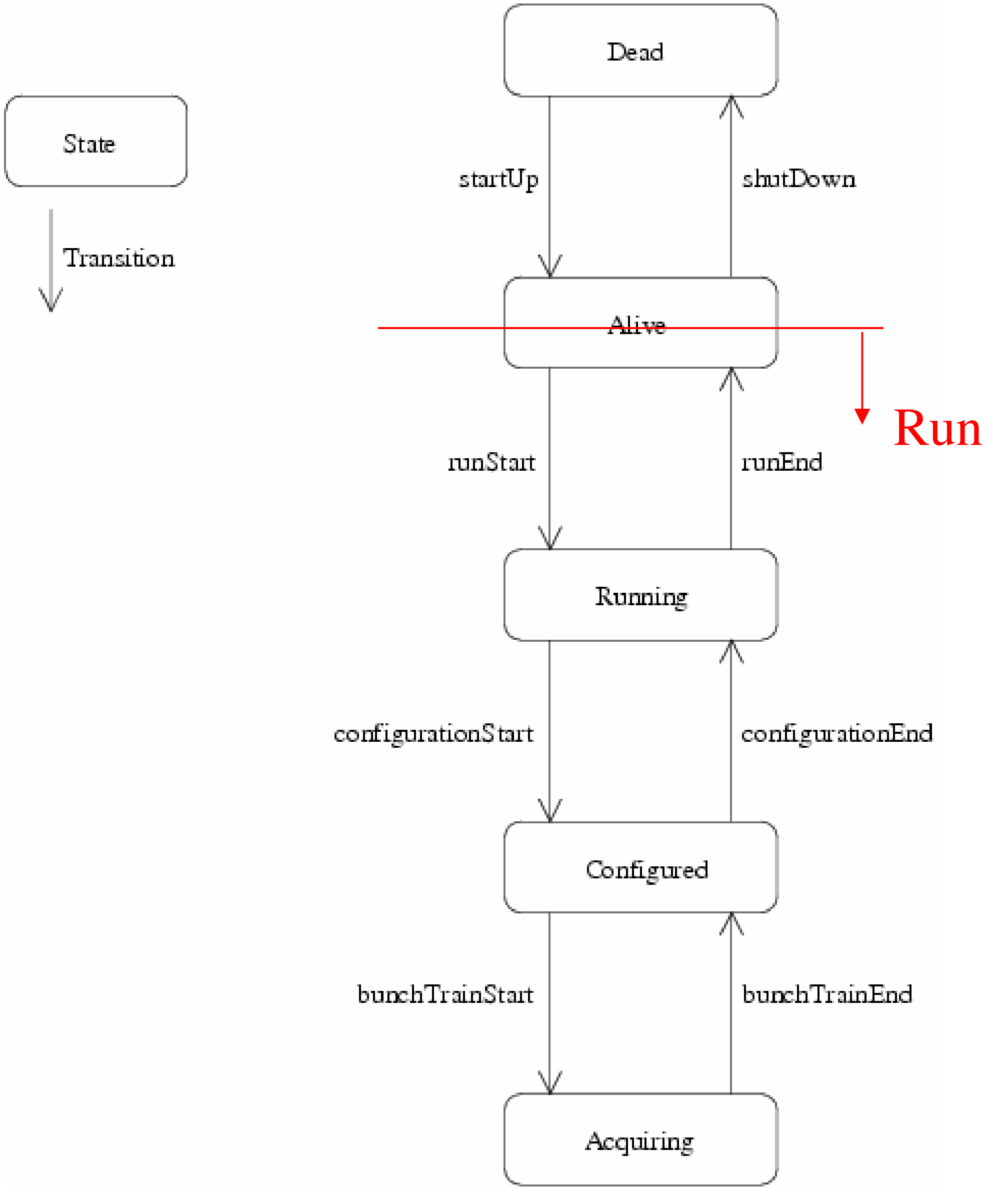
With Jamie Ballin and Matt Noy

DAQ overview

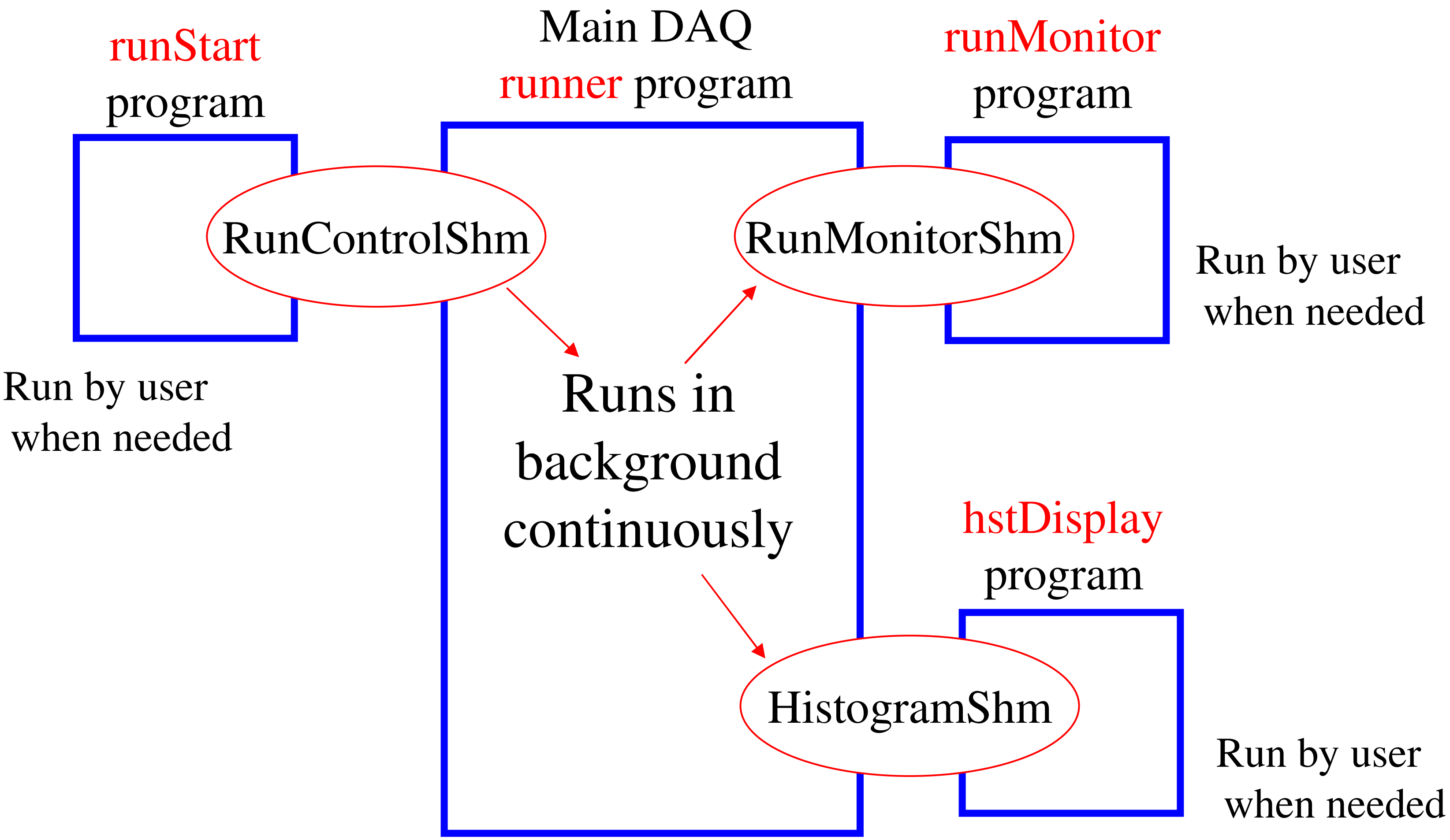
- Like most DAQ systems, based on a state machine
  - Transitions between states driven by transfer of “records”
- Records are both dynamical agents and data storage elements
  - Record type tells each node of system which state to go to
  - Record contains data needed by node to complete transition…
  - …and/or stores data generated by node during transition
- E.g. configurationStart record
  - Tells a node to configure some hardware
  - Contains configuration values to do this
  - Stores the configuration values read back by node as crosscheck
- Records are simple contiguous arrays in memory
  - Written to (and read from) “binary” raw data files
- All configuration data written into the records
  - No need for database for basic offline analysis

A run is set of nested levels of transitions

- runStart
    - configurationStart
        - bunchTrain
        - bunchTrain
        - bunchTrain
        - …
    - configurationEnd
    - configurationStart
        - bunchTrain
        - bunchTrain
        - bunchTrain
        - …
    - configurationEnd
    - …
- runEnd



Executable structure



Software overview

- Runs are labelled by runTypes
  - E.g. mpsNoise, mpsBeam, mpsLaserPositionScan, etc.
  - Each has a version number to select choices, e.g. number of x,y points in scan
- Together, runType and version determine
  - Number of configurations in run
  - Number of bunch trains in each configuration
- No databases currently used
  - There is a lack of widespread knowledge
  - Could be included as part of configuration data loading if we want
- Code requires
  - Any standard C++ compiler on standard Linux PC
  - ROOT
  - USB_DAQ driver

Existing runTypes

- mpsTest, mpsExpert, mpsNoise
  - Basic runs for setting up
- mpsConfigurationTest
  - Takes no data; just does configuration load
- mpsThreshold, mpsThresholdScan, mpsTrim, mpsTrimScan
  - Scans thresholds
- mpsBeam, mpsCosmics, mpsSource
  - For main data-taking
- mpsLaser
  - Like mpsNoise but with the laser
- mpsLaserThreshold, mpsLaserThresholdScan mpsLaserPosition, mpsLaserPositionScan
  - Scans of threshold and position with laser

To be done

- Hardware-level
  - Real connection to the USB_DAQ
  - Socket connection to the laser
- Online software-level
  - Precise definition of data formats
  - More realistic configuration values
  - Definition of further run types
  - GUIs for run control and monitoring
- (Semi) offline software-level
  - Conversion of raw data to ROOT trees
  - Histogramming and GUI
- Analysis software
  - Pretty much everything
  - Simulation; do digitisation process through DAQ?