The *artdaq* Data Acquisition Toolkit

Kurt Biery, Chris Green,
Stephen Foulkes, James Kowalkowski
Fermilab Scientific Computing Division

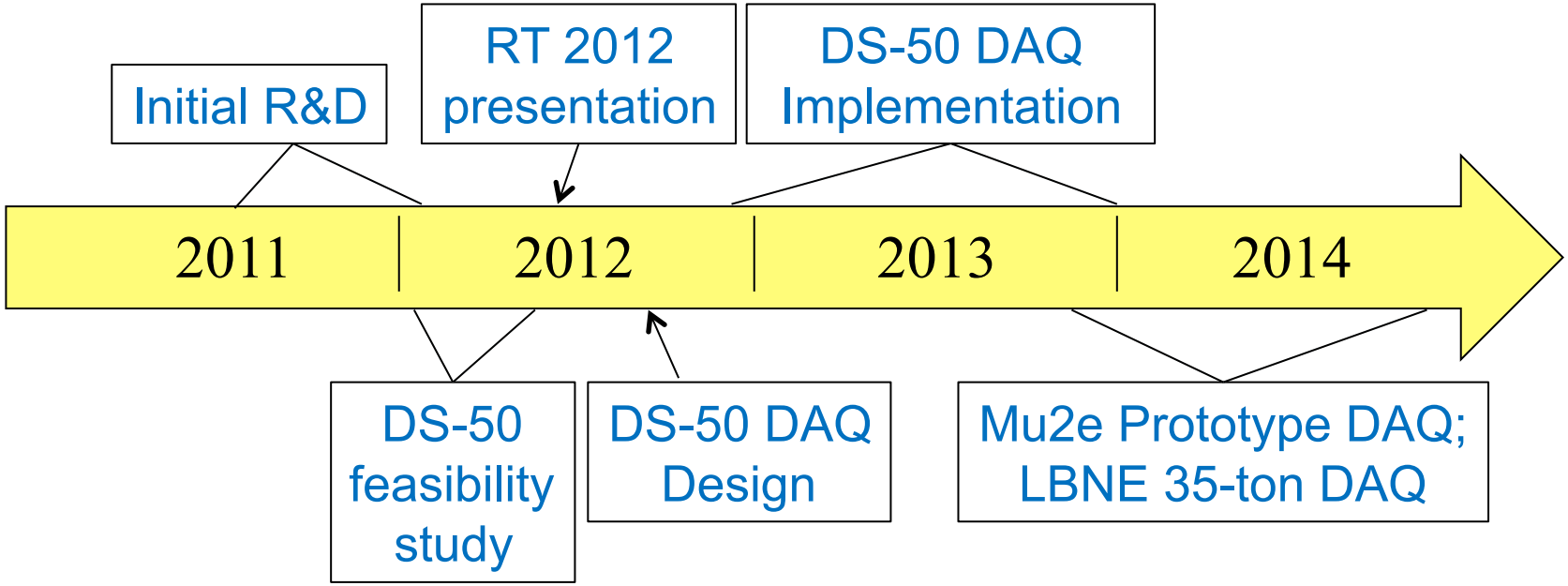
CHEP October 2013

What is *artdaq*?

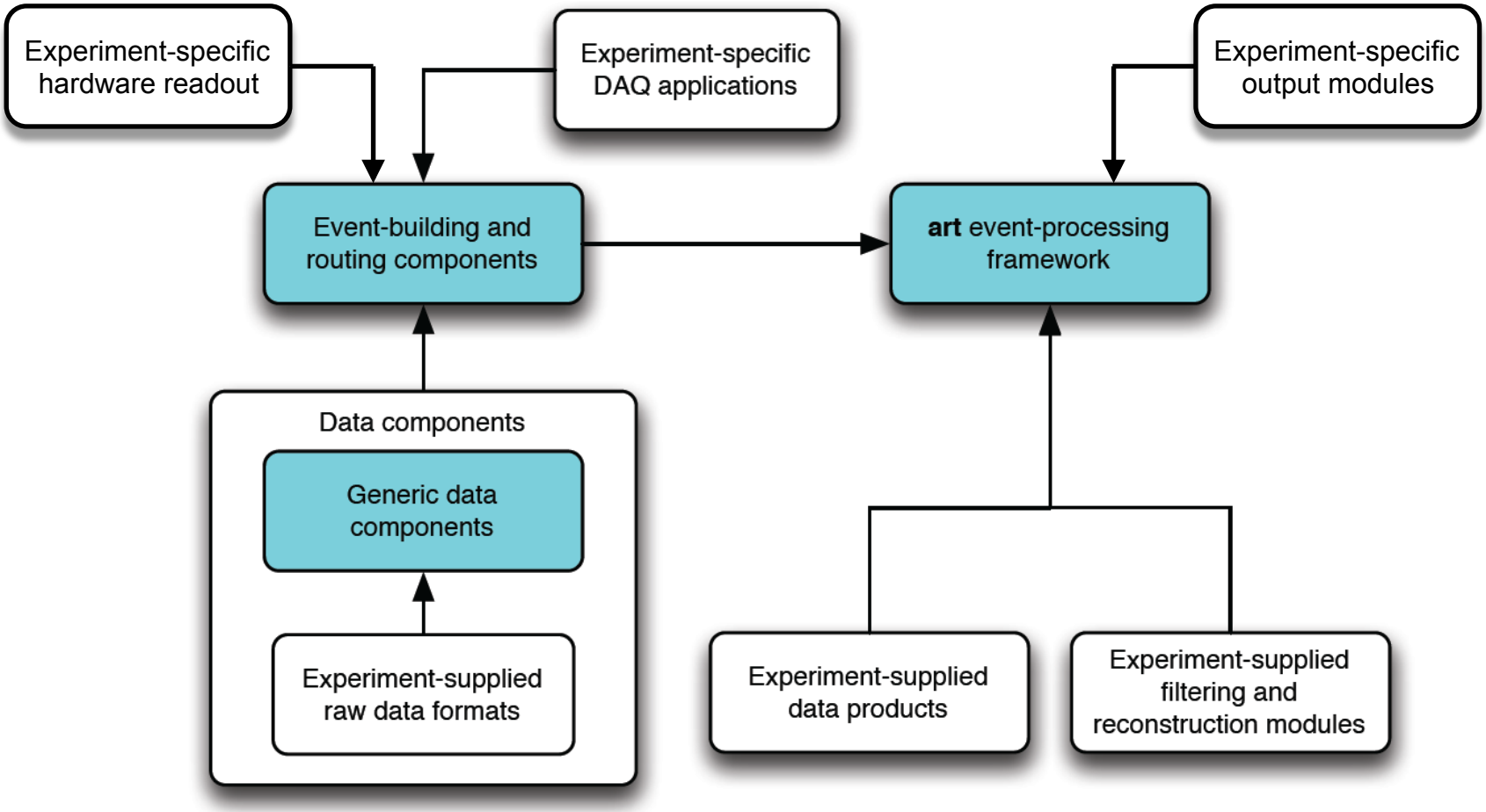
artdaq is a toolkit for creating data acquisition systems

- Fermilab Scientific Computing Division's core DAQ software
  - Provides common, reusable components
  - Based on a data-streaming architecture with software event filtering
- Integrated with the *art* framework
  - Same algorithms online and offline

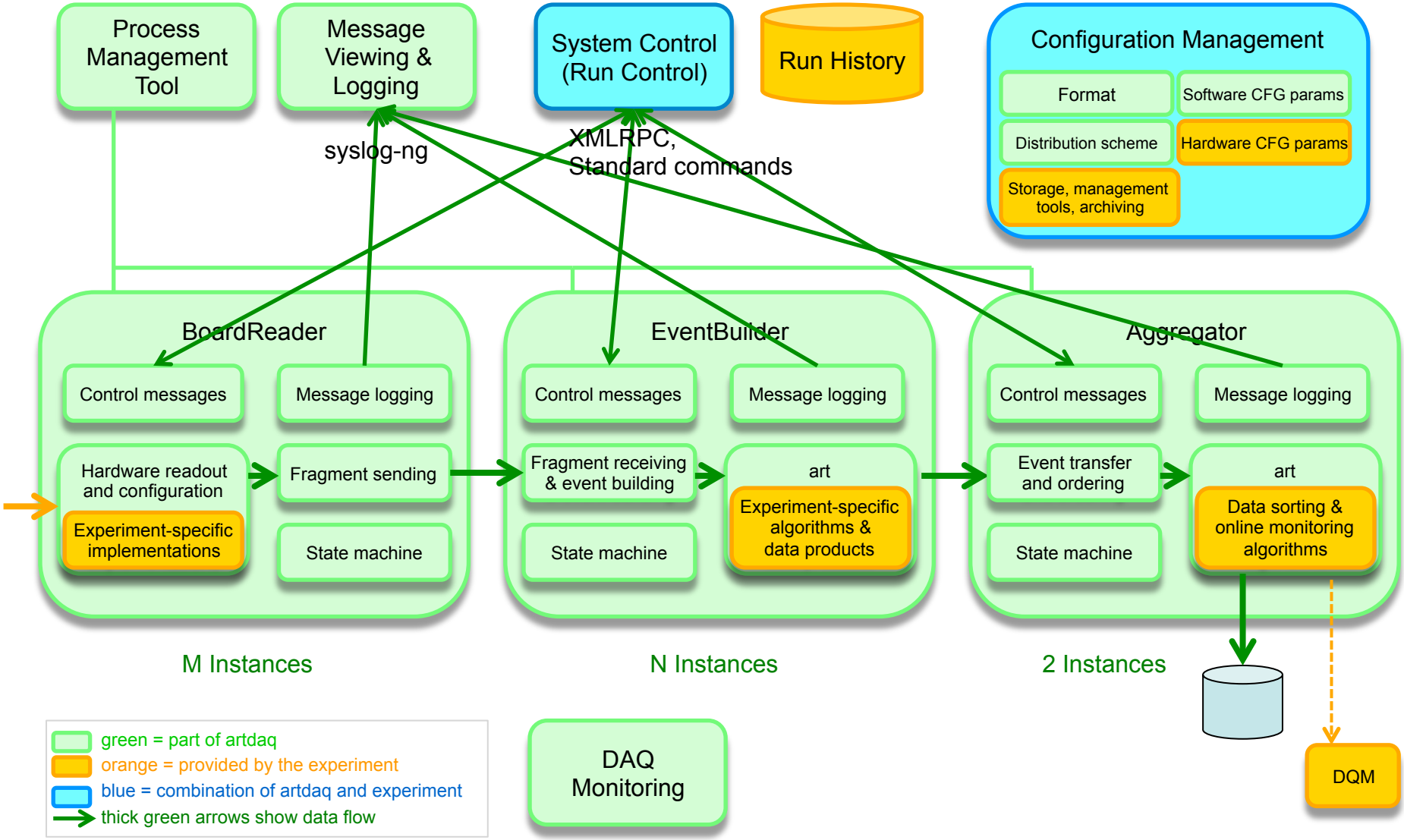
Timeline


Initial R&D
RT 2012 presentation
DS-50 DAQ Implementation
2011
2012
2013
2014
DS-50 feasibility study
DS-50 DAQ Design
Mu2e Prototype DAQ; LBNE 35-ton DAQ

Initial *artdaq* Architecture



artdaq Software Components



artdaq for DarkSide-50

DarkSide

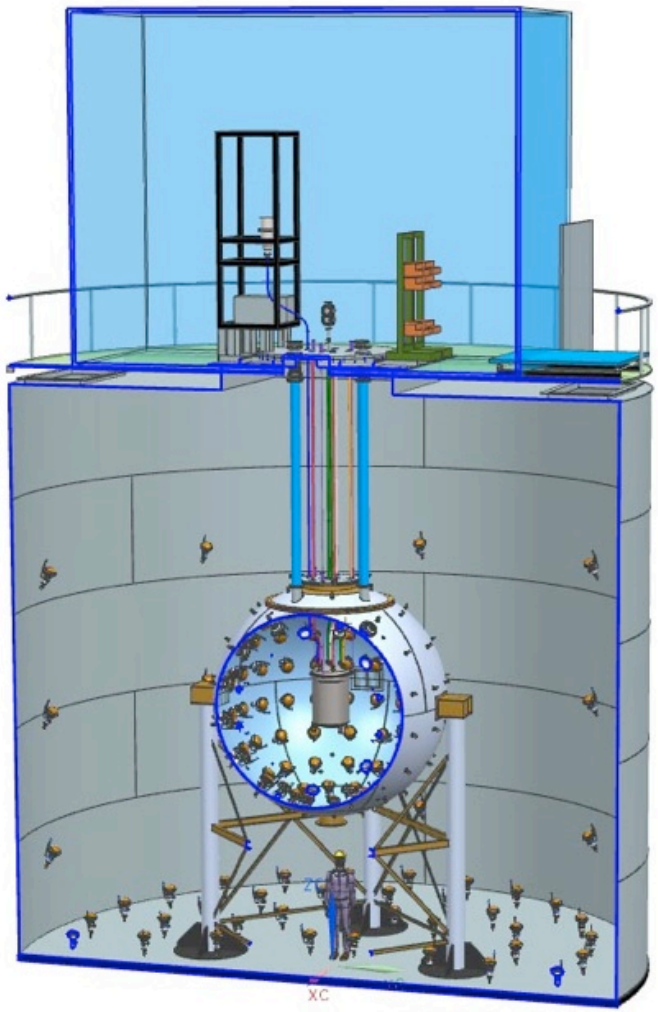
- Direct search for WIMP Dark Matter
- Liquid argon detector
- Underground at LNGS
- Part of Fermilab Cosmic Frontier program

Timeline

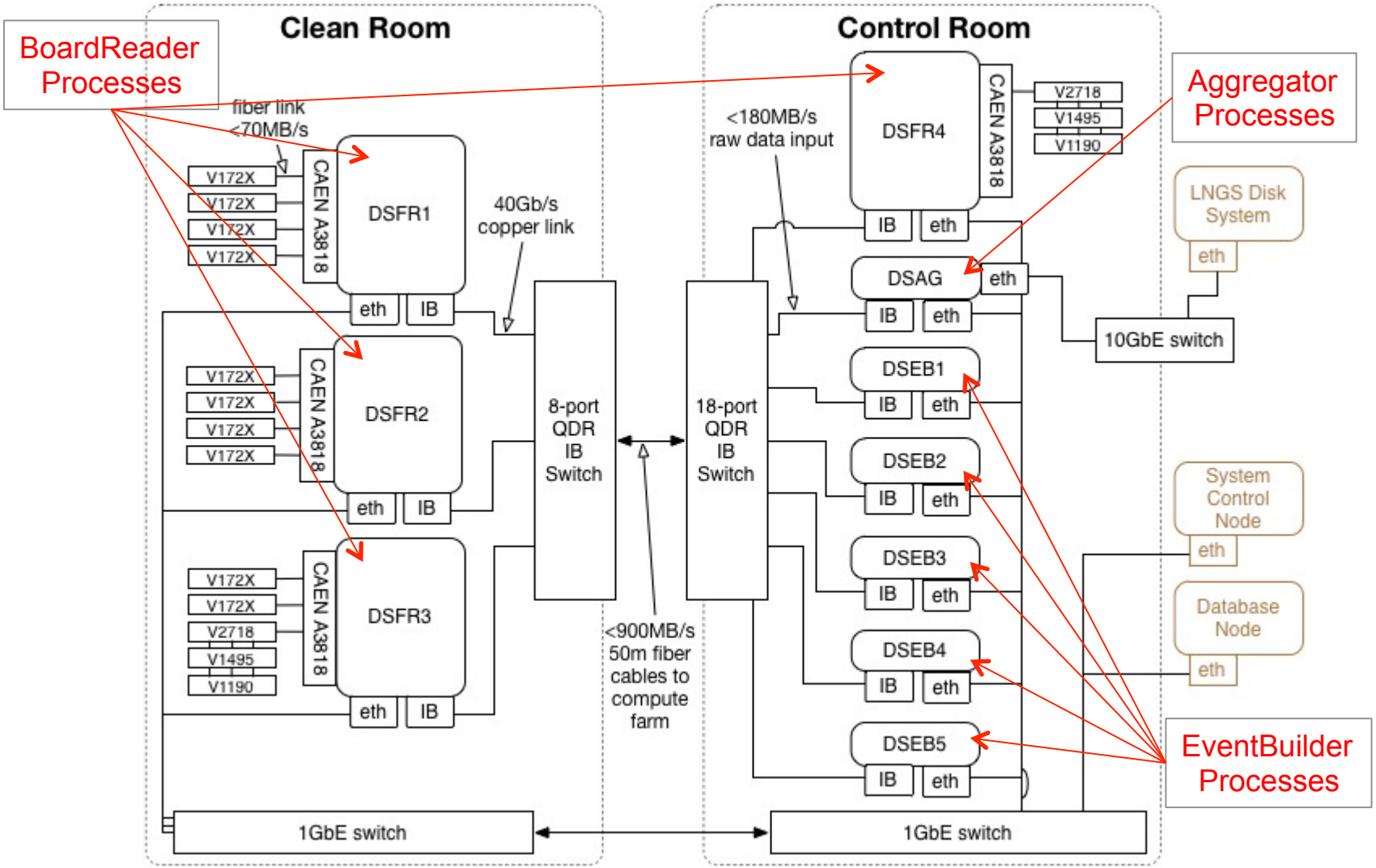
- Detector and DAQ commissioning – now
- Start of 3-year data taking – early 2014

DAQ

- Commercial VME modules (CAEN)
- Commodity servers and networking
- *artdaq* software (first production use)
- Typical event size ~ 10 MB
- Event rates up to 80 Hz



artdaq for DarkSide-50



artdaq for DarkSide-50

Collaborative Development

- Fermilab and LNGS staff
- December 2012 – now

Additions to core infrastructure

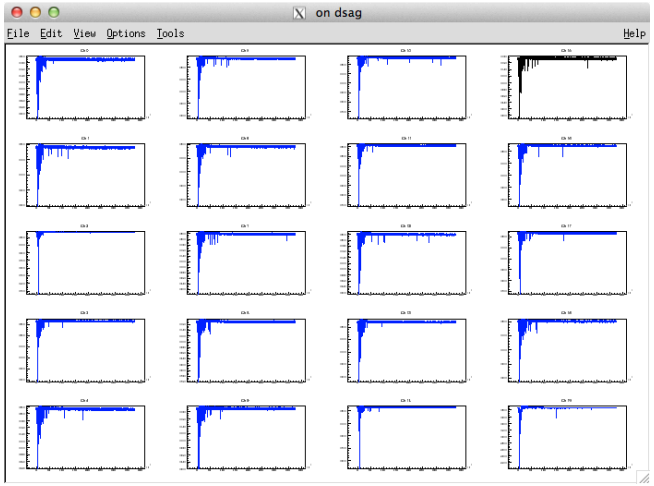
- Aggregator to sort and store events
- *art* enhancements to support online use

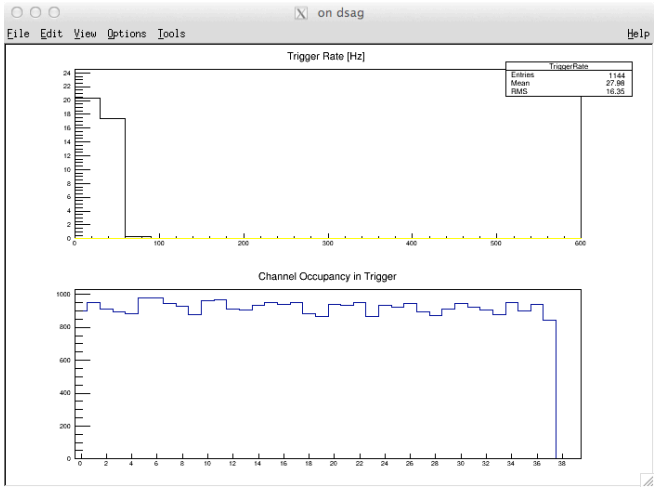
Customizations for Dark-Side

- Configuration and readout of VME modules
- Loss-less data compression
- Online monitoring

Status

- Reliable operation
- Meeting performance needs

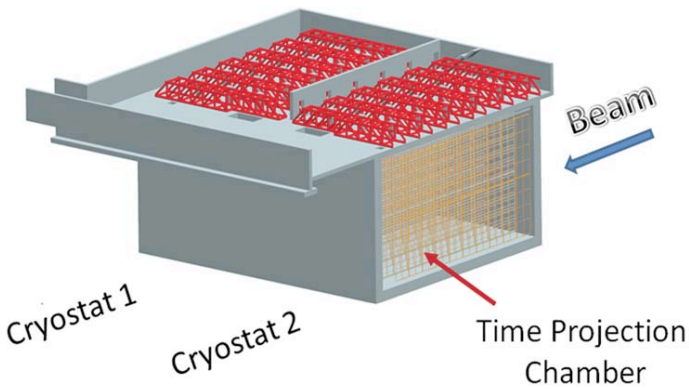




artdaq for LBNE

Long-Baseline Neutrino Experiment

- Neutrino oscillation measurement
- Liquid argon detector
- Neutrino beam Fermilab -> Sanford lab in South Dakota

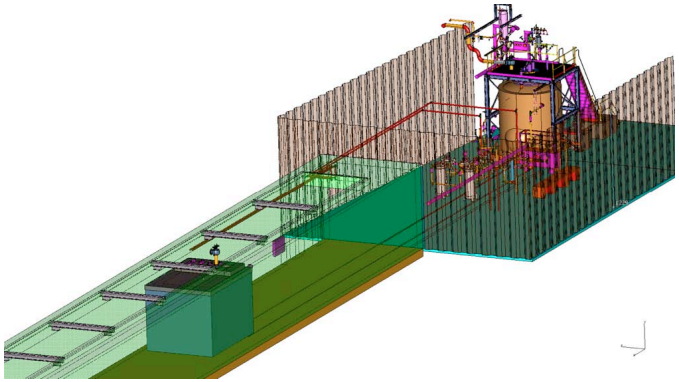


Timeline

- 35 ton prototype – early 2015
- Full detector – 2024

DAQ

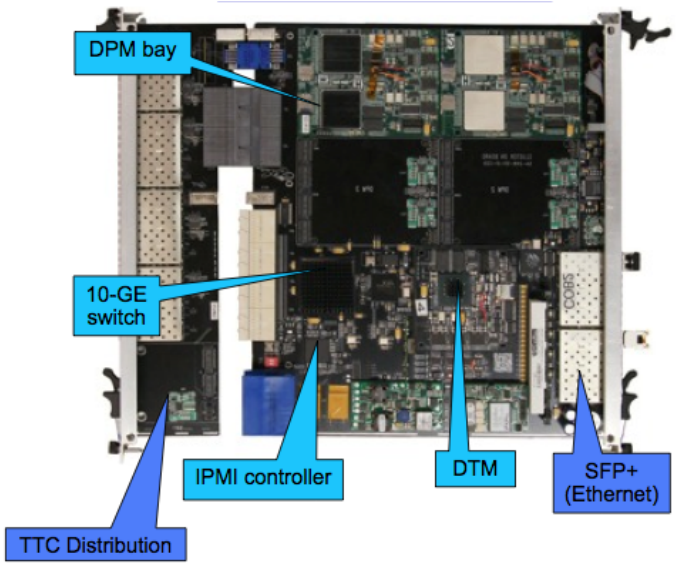
- Custom front-end electronics
- SLAC RCEs for front-end readout and possible zero-suppression
- *artdaq* software
- Data rate ~4 GB/s
- 2400 connections to the front-end electronics



artdaq for LBNE

Preparations for 35t detector

- Collaboration of engineers, developers, and scientists from Fermilab, SLAC, and university groups
- Integrated readout of TPC (SLAC RCE) and photon-detection system (CAEN VME) with *artdaq* software



Integration with existing components

- Run Control from IceCube
- Configuration management from NOvA

LBNE-specific development

- Porting of *artdaq* to RCE
- Reconstruction algorithm development and testing

artdaq for Mu2e

Mu2e

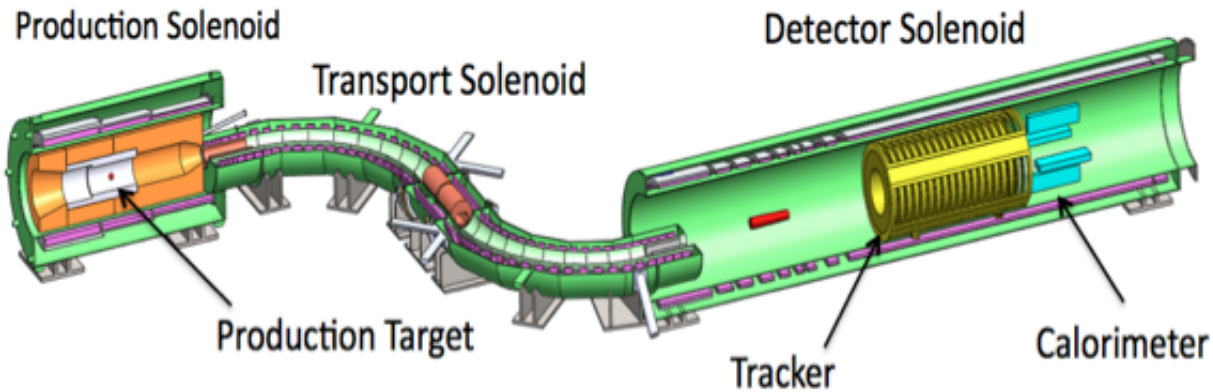
- $\mu^- N \rightarrow e^- N$
- Stopped muons in Al target
- Part of Fermilab muon campus

Timeline

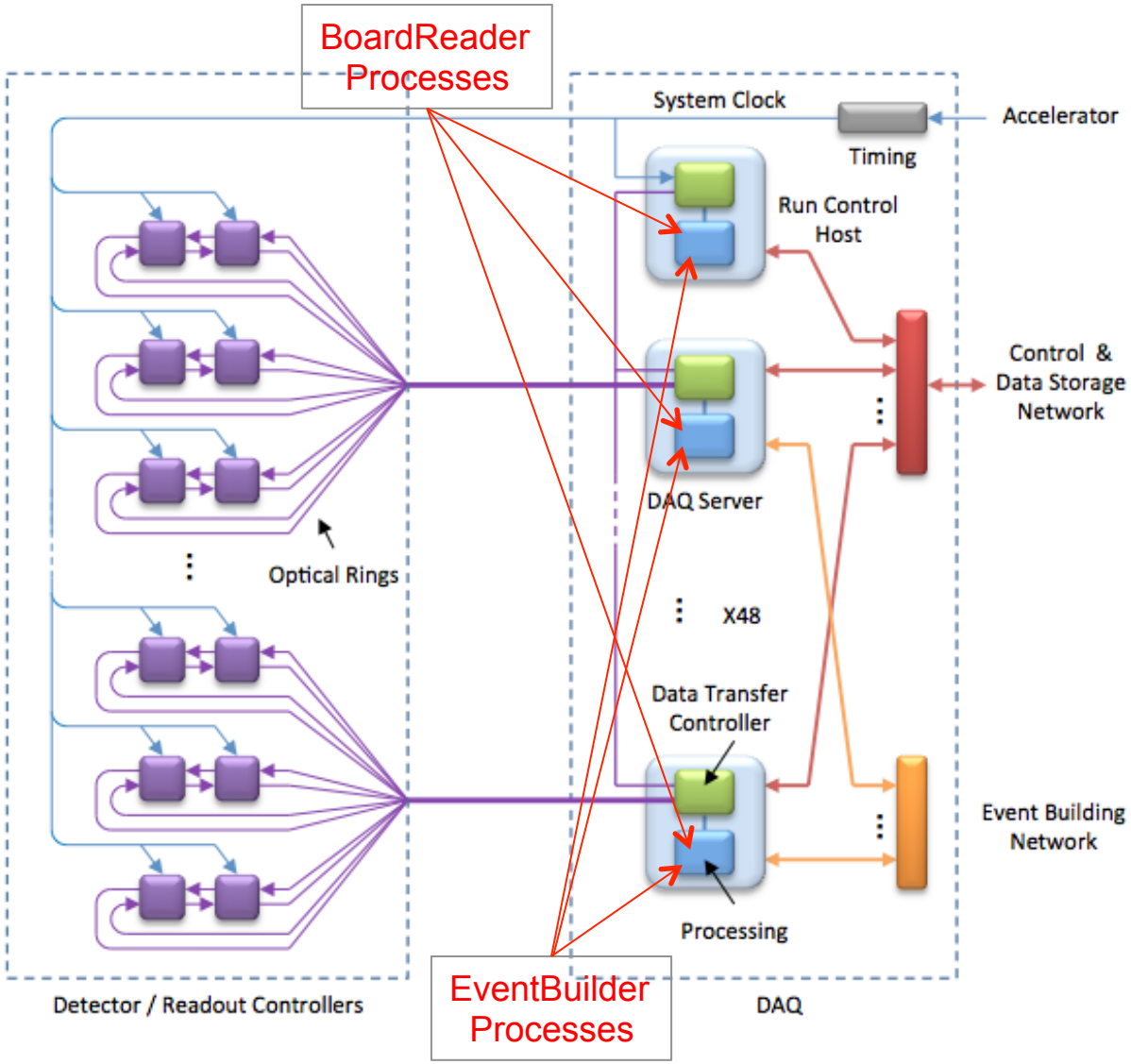
- CD-2/3 approval July 2014
- Commissioning data in 2019

DAQ

- Custom readout electronics
- Commodity servers and networking
- *artdaq* software & custom firmware
- Expected data rate: 30 GB/s
- Software rejection factor: 1000



artdaq for Mu2e



artdaq for Mu2e

Preparations for CD-2/3 review

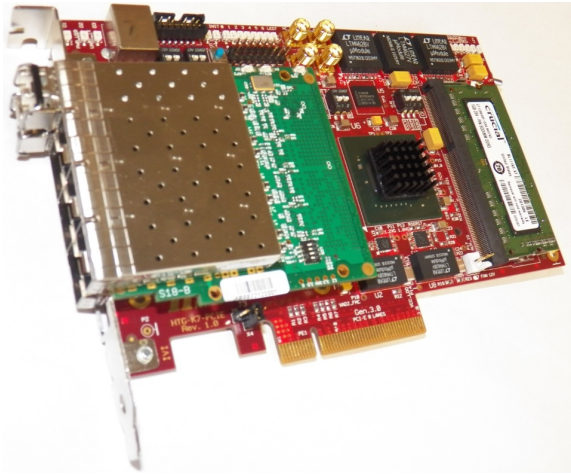
- Fermilab engineers and developers
- Readout of commercial PCIe cards with *artdaq* by December 2013

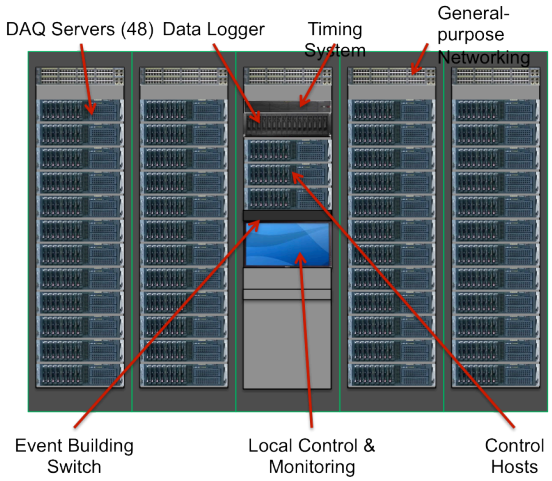
Possible additions to core infrastructure

- Run Control GUI
- Configuration management tools

Mu2e-specific development

- Configuration and readout of hardware
- Reconstruction algorithm development and testing





Some Technical Details

Uses C++11 features

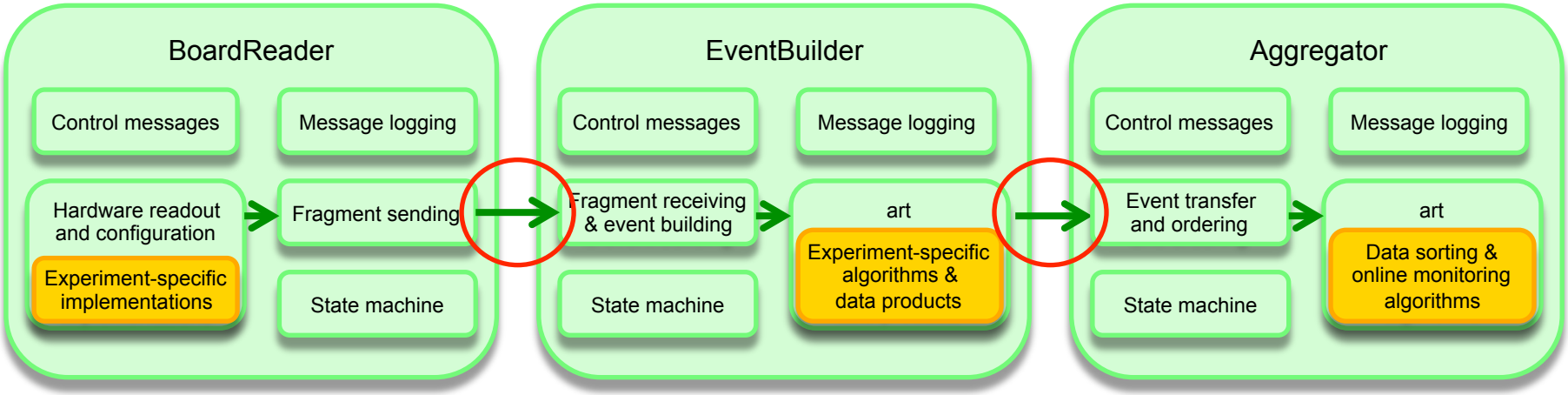
- e.g. move semantics to minimize data copies

MPI for data transfer

- Wrapper classes for sending and receiving MPI buffers

Process management

- Wrapper script around *mpirun* command



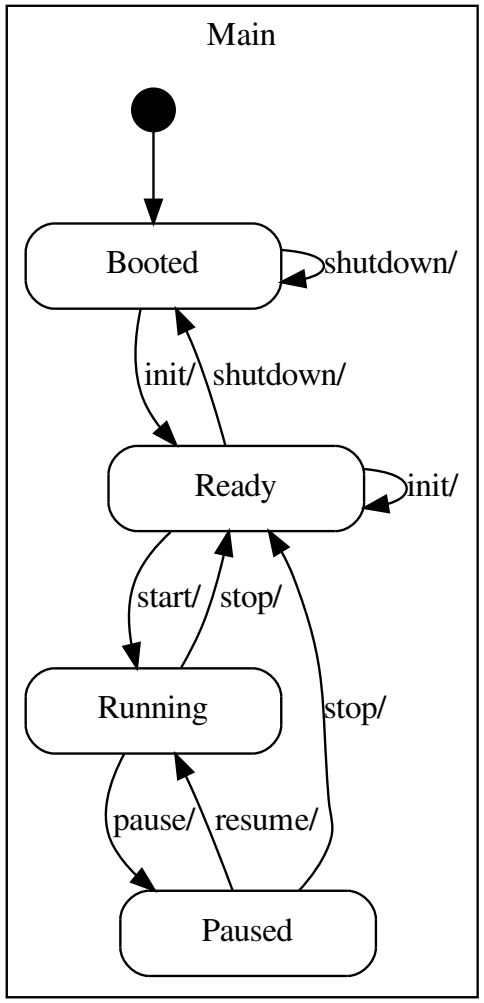
State Behavior & Control Messaging

State behavior part of *artdaq* core

- Standard states:
  - Booted, Ready (configured), Running, Paused
- Standard transitions:
  - Init(cfgString), start(runNum), stop, pause, resume, shutdown
- State Machine Compiler tools
- Overall system state managed by Run Control

Control messaging infrastructure also included

- Standard commands:
  - State transitions & status queries
- XMLRPC
- Command-line Run Control scripts provided



Configuration Management

Configuration infrastructure part of core *artdaq*

- Single structured text string sent at “init” transition
  - hardware, software, system, *art* configuration
- Interpretation of software and system parameters provided in core *artdaq*
- Hardware parameters handled by experiment-specific code
- Archiving and management of configuration data – not yet part of core *artdaq* – but may be in the future

```
daq: {  # red = group identifiers
  max_fragment_size_words: 2097152
  fragment_receiver: {
    mpi_buffer_count: 40
    generator_ds50: {
      fragment_id: 2
    }    # green = software params
    generator: V1495Driver
    first_event_builder_rank: 3
    event_builder_count: 5
    rt_priority: 2  # system params
    link_type: PCIE
    link_number: 0 # hardware params
    vme_base: 0x01000000
    pulser_frequency: 0.0
    laser_frequency: 0.0
    random_triggers: false
    acquisition_gate_us: 400.0
    low_threshold: 5
  }
}
```

Typical Analysis Tasks (*art*)

Data compression

- DarkSide-50 – factor of 5-6 reduction

Online monitoring

- DarkSide-50

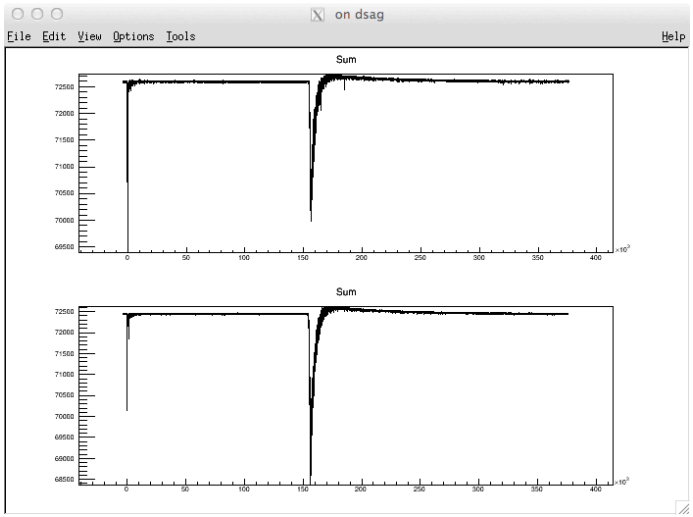
Event filtering

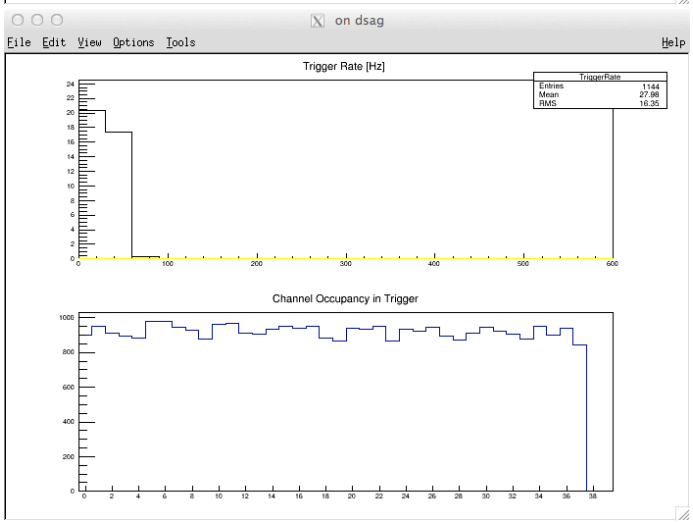
- Mu2e – reduction by factor ~1000

Event selection

- NOvA – data driven trigger

Same code online as offline

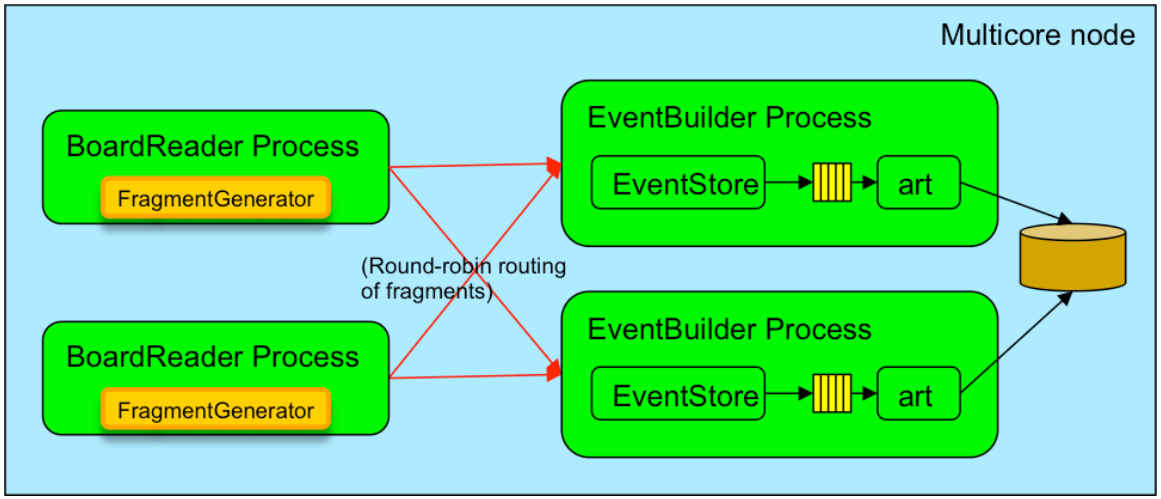




The artdaq-demo

Demo package to illustrate *artdaq* use

- Instructions for downloading, building, and running a sample system
- More information here:
  - https://cdcvs.fnal.gov/redmine/projects/artdaq-demo/wiki
- Updates to software and instructions coming soon



Additional Users

NOvA DAQ

- *art* for enhanced triggering (data-driven triggers)
- Minor pieces of *artdaq* for *art* input

uBooNE

- Uses some *artdaq* utility classes

Fermilab testbeam

- *artdaq* being discussed for Lariat, etc.

Plans

Medium-term

- Centralized message logging
- Cross-compiling for ARM processor (SLAC RCE)
- Performance improvements

Longer-term

- Improved online monitoring scheme
- Run Control GUI
- DAQ monitoring – statistics, stripcharts, etc.
- Alternate data transfer technologies
- Alternate control message technologies

Alternate technology options for various functions will be driven by future experiment needs