Data Acquisition Systems for Experimental Nuclear Physics

by David Abbott
Introduction

About me:
- Ph.D UNC Chapel Hill (1990)
- JLAB 1991-now
- With the Data Acquisition Group since 1994

About us:
- 6 member Physics support group
- Cebaf Center: F-Wing
What is this talk about...

- What is Data Acquisition?
- The anatomy of a DAQ system.
  - DAQ architectures.
  - What do all the bits and pieces do?
  - What does it all look like?
- What are we doing here at JLAB?
- Show and Tell along the way…
What is Data Acquisition?
What is the goal?

- The aim of a nuclear physics experiment is to gather data about nuclear interactions.
- Nuclear particles pass through detectors which generate electrical signals.
- These signals contain information about the particles - type, energy, trajectory.
- The data acquisition system digitizes, formats and stores this information in a way which can be retrieved for later analysis.
What are the problems?

- The complete set of signals which describe a single nuclear interaction is called an Event.
- There can be thousands to millions of events occurring per second.
- Detectors are large and distributed - containing many thousands of individual channels.
- Events are different sizes.
- Events occur at random.
- Only a few events are interesting.
Data Acquisition Requirements

- Move the data: Detector --> Storage
- Configure and control experiments
- Monitor data flow
- Monitor detectors/hardware
- Inform operator of problems
- Experiments can run for days/weeks/months…
The Anatomy of a DAQ System.

- **Triggering** (choosing events we want)
- **Readout** (digitizing detector signals)
- **Event formatting** (standardize what we’re saving)
- **Event building** (putting fragments together)
- **Event transport** (make events available to all)
- **Event storage** (save data for analysis)
- **Run Control** (configure-start-stop experiments)
- **Monitoring** (tell me what’s going on)
- **Slow Controls** (What is the other hardware doing?)
A DAQ System example

- Digital Camera is a “simple” physics DAQ system.
- 3-6 million channels
- *Dead-time* is important!
  - How long before I can take another picture??
- DAQ requires a “Real-time” response.

CCD detector

Readout, processed

Light

Data stored on flash card

Trigger
Triggering

- The data acquisition system needs to know when an interaction “Event” has occurred in the detector.
- Some detectors are faster than others.
- Signals from fast detectors are combined to make a decision on when an event has occurred. This is called a trigger.
Triggering cont...

- Triggering serves two functions:
  - Tells the rest of the system when to read.
    - Trigger tells DAQ to read out data
    - DAQ tells trigger when it is busy
    - Busy time is called dead-time, and is minimized by a well-designed DAQ architecture (see later).
  - Filters unwanted events.
    - Most triggers work in levels.
      - Level 1 is based on fast detectors like scintillators.
      - Level 2 is based on slower detectors (drift chambers).
      - Level 3 is usually a software filter.
A Simple Trigger

A "coincidence" occurs if two logical pulses overlap - Typically within about .000000020 seconds.
Hall A Trigger
What it really looks like.

Hall B
CLAS
Trigger
Readout

- Data takes the form of electrical signals.
  - Convert Analog --> Digital
  - Times - Time to Digital Converter, TDC
  - Voltages - Analog to Digital converter, ADC
  - Counts - scalars

- There are lots of signals spread over a large detector.
  - Modular readout duplicated many times
  - Plug-in modules require something to plug into so that they can all be accessed together --> Buses
Digitizing Data

Detector data must be digitized to be stored on a computer. When a particle passes through a detector, we can measure the amount of energy (charge) it leaves (ADC), or we can measure the time it takes to travel between two different detectors (TDC).

The digitizer makes a 12bit number (0 - 4095) that is proportional to the amount of charge (or time). The physicist must calibrate to digitizer so that he knows the energy (or seconds) each number represents.

<table>
<thead>
<tr>
<th>Digitized Value</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1520</td>
<td>50 MeV</td>
</tr>
<tr>
<td>3040</td>
<td>100 MeV</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Digitized Value</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>220</td>
<td>5 nanoseconds</td>
</tr>
<tr>
<td></td>
<td>(0.000000005 seconds)</td>
</tr>
</tbody>
</table>
Detectors in the Experimental Halls have many thousands of channels. Each Channel is read (digitized) by an ADC or TDC.

- Pack many circuits onto one board.
- Pack many boards into a box.
- Pack the boxes into racks.
- Use a standard bus to link everything.

Standards: CAMAC, FASTBUS, VME and PCI.
CAMAC

(Computer Automated Monitoring And Control)

- Old IEEE Standard
- 24 bit bus.
- Relatively slow (3 MB/sec).
- Small boards.
- A lot still around.
- Slow controls
FASTBUS

- Designed by physicists for physicists.
- Large form-factor, high channel densities.
- 32 bit bus (40 MB/sec)
- Majority of JLAB detectors interfaced.
- No more commercial Vendors
Full Crate
VME (Versa Module Europa)

- International standard for interconnecting modules.
- 32/64 bit bus (80MB/s)
- Large number of commercial products (used heavily in the military).
- VME64X provide bandwidth options (160-320 MB/s).
- Currently transitioning from FASTBUS
SFI FASTBUS to VME Controller
**PCI (Peripheral Component Interconnect)**

- It’s in every PC you buy today.
- Fast 64 bit bus (33-66MHz, 266 - 532 MB/s).
- Not much specialization for nuclear physics.
- Small board size.
- Different “flavors”
  - PCI-X (133 MHz)
  - cPCI (Bus “module” format)
  - PMC (daughtercard)
  - PCI-Express (PCIe serial)
Interface to Detectors

- The Trigger and ADC "Gate" begin the conversion and readout phase
- Fast (real-time) response to trigger is important to minimize dead-time
HMS Detectors

Scattered Beam

Phototubes

Drift Chambers
Hall A Spectrometer
The Real World
... and it gets worse!
Event formatting

- The data comes from different detectors.
  - Need to identify the detector
  - Need to identify which event this came from
  - Need to make analysis easier.
- There is a lot of data
  - Format must be compact
- Analysis can take years
  - Format should be *self documenting*
What's in a Number?

For each detector channel a number is generated when a trigger occurs. There may be many thousands of detector channels for a given physics experiment, and there may be many thousands of triggers generated per second. There is the potential for a huge amount of data...

Physicists must be very efficient in the storage of information. So what does typical detector TDC data look like?

968,033,935

1 number (value between 0 - 4.2 billion)

4 Bytes of Data on a disk

32 bits of Data

16 bits for time to an accuracy of 0.0000000006 seconds.
In this case: 1679 => 100.74 ns => 0.00000010074 seconds
(This is the time between a trigger and a hit in this detector channel)

3 bit chip address (0-7)
In this case: 6

3 bit channel address (0-7)
In this case: 3

2 bit Data Identifier
In this case: 2 (TDC data)
Example format

Pack data in “banks” and provide “layering” of information

4 Bytes each
Data flow

- Once all the data has been digitized it must be collected into a central place for storage.
- How the data is moved from the detector readout to the storage medium depends on several factors.
  - Available technology
  - Event size and trigger rate
  - Your budget!!
  - Personal taste
- More in DAQ architecture section later.
The Physics data flows on…

Scintillator
Phototubes
Discriminator
Coincidence
Event Building
Analysis
Storage
Readout Controller
ADC
Single Board Computer
Real Time OS
Trigger
Gate
Network
UNIX OS
Event Building Analysis Storage
Event Building

- The detectors are spread over a physical volume of space.
- Bits and pieces of events arrive at different times from different places.
- All the parts of the event need to be collected together and packaged with other information needed by the analysis.
- The *Event builder* is a very fast collating machine.
CODA “Push” Architecture

Detectors

ROC

Write Thread

FIFO

Read Thread

Read Thread

Read Thread

Real-Time OS

Network

Build Thread
Event storage

- Since the goal is to store data we need somewhere to put it.
- Physics experiments generate a lot of data. At JLAB 2-35 MB/s per Experiment.
- The fastest method is to Disk!!
- The most cost effective method is tape.
- Stage data to disk -> then backup to tape.
- The tape drives must be fast (and robotic). Fast tape drives are expensive!!
- Aim for tape to be the limiting factor in DAQ speed.
Tape Silos
Storage at JLAB

From Halls

Mass Storage 2PB

DST/Cache FileServers 29 TB

Jefferson Lab
Farm and Mass Storage Systems
January 2006

Farm Cache FileServers 5 TB

DB Server

Work FileServers 29 TB

From CLAS DAQ
From Hall A,C DAQ
100 mbit
1000 mbit
PCAL (100MByte)
SCSI

Batch and Interactive Farm
~360,000 SPEC CINT2000
Run Control

- Need to start and stop the DAQ
- Place to input parameters which change from run to run.
- Place to read parameters from.
- Automatic monitor of the health of the DAQ system.
- Something nice for the operator to look at.
CODA, the Jefferson Lab Common Data Acquisition System

There is page tells you something about this program, RunControl. Help for the rest of CODA can be found [here](#).

About this program

This is the main control panel for CODA, the common data acquisition software, for all experiments at Jefferson lab. Since you are reading this page you have already pressed the 'reset' button.

The control part of the interface

Menu Bar

To the left of this page is the main control panel. At the top of the panel is a menu bar which allows you to select various options and shutdown the GUI. There are two programs involved in controlling...
JAVA Run Control
Monitoring/Analysis

- Need to monitor the data quality as it is read.
- Interface between code written by Physicists and code written by DAQ experts.
- Primary goal is to distribute data to anyone who needs it.
- Monitoring must not introduce dead-time.
- CODA Event Transfer “ET” System
Slow Controls

- A topic all by itself.
- Covers all of the other data about the experiment which needs to be acquired.
  - Power supply voltages.
  - Magnetic fields
  - Beam position
  - Target position
  - Vacuum pressure
  - Coffee Maker Status …
The real world, again
Experiment Control

- CODA Java-Based (v 1.5) “Intelligent” agents
- JADE extensions provide a runtime “distributed” JVM.
- Agents provide a customizable intelligence and communication with external processes.
DAQ architectures

Given all the parts of a DAQ system how are they put together.

Architecture depends on

- Event rate
- Event size
- Trigger type
- Available technology
Examples
Older DAQ Systems

Circa 1980s - DAQ systems were closed and custom-built based on the detectors.

Single Mainframe CPU systems processed the data.
Break out of a closed system and Rise of the Network (1980s-90s)
LHC - at CERN (2007->…)

14 TeV Collider
>16 mile circumference
4 primary experiments

- ATLAS
- CMS
- LHCb
- ALICE
CMS Trigger and DAQ

Data Acquisition Main Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collision rate</td>
<td>40 MHz</td>
</tr>
<tr>
<td>Level-1 Maximum trigger rate</td>
<td>100 kHz</td>
</tr>
<tr>
<td>Average event size</td>
<td>1 Mbyte</td>
</tr>
<tr>
<td>No. of electronics boards</td>
<td>10000</td>
</tr>
<tr>
<td>No. of readout crates</td>
<td>250</td>
</tr>
<tr>
<td>No. of In-Out units (200-5000 byte/event)</td>
<td>1000</td>
</tr>
<tr>
<td>Event builder (1000 port switch) bandwidth</td>
<td>1 Terabit/s</td>
</tr>
<tr>
<td>Event filter computing power</td>
<td>5 x 10^6 MIPS</td>
</tr>
<tr>
<td>Data production</td>
<td>Tbytes/day</td>
</tr>
</tbody>
</table>

Trigger and Data Acquisition baseline structure

16 Million channels
3 Gigacell buffers

1 Megabyte
EVENT DATA

200 Gigabyte
500 Readout buffers

EVENT BUILDER.
A large switching network (512x512 ports) with a total throughput of approximately 1000 Gbit/s forms the interconnection between the sources (Readout Dual Port Unit, RU) and the destinations (Filter Unit, FU). The Event Manager collects the status and request of event filters and distributes event building commands (read/clear...) to RDPms.

5 TIPS (5 x 10^6 MIPS)
500 CPU farm

EVENT FILTER
It consists of a set of high performance commercial processors organized into many farms convenient for on-line and off-line applications. The farm architecture is such that a single CPU processes one event.

Petabyte ARCHIVE
What is this CODA stuff?

- **CEBAF Online Data Acquisition**
- CODA is a software toolkit with some specialized hardware support.
- Modular software components use the network for inter-process communication and event transport.
- Use *open standards* and minimize the use of commercial software while maximizing use of commercial hardware.
- DAQ systems for each experimental Hall can be “built-up” from common components to fit their needs.
CODA is a software toolkit from which data acquisition systems with varying degrees of complexity can be built.

A typical system might look
What next?

- Interesting Physics becomes more experimentally difficult ("good" events are more rare)
- Current DAQ is reaching "Limits" of performance
- Technology is always changing.
  - FASTBUS is DEAD as a standard.
  - Computer hardware becoming faster (CPU, RAM, NET)
  - More can be done in software (Real time moves to HW)
  - Busses have reached limits - Hi-speed (2.5-10 Ghz) Serial/Fiber.
    - (PCI-X 133MHz ~8Gbit/s    16x PCIe (2.5Ghz) -> 40Gbit/s)
- Reduce dependence on operating systems.
  - Ultrix -> HP-UX -> Solaris -> Linux ====> JAVA
- "Customizable" hardware is becoming a viable option (FPGAs, DSPs, ASICs).
Sampling vs Integration

Traditional “integrating” ADC takes 6-10 µsec to digitize

250 MHz Flash ADC samples every 4nsec

Generates ~10-15 data words that describe the pulse

Generate 1 word representing the charge sum during the gate.

We can also use these samples to generate the Trigger...
Pipelines (dead-timeless DAQ)

10μs “snapshot” can be stored in memory (~5KB/ADC)

Trigger a look-back and select relevant data.
Pipelines (dead-timeless DAQ)

Phototube

Scintillator

FADC

FADC

FPGA

Trigger

Sums & Hits

Trigger From Sums

VME

FPGA

5

4

3

2

1
JLAB Flash ADC

250MHz FADC ASICs

FPGAs

fADC250 @ 250 MSPS
Pulse on CH 1: 30 MHz, tr & tf = 5 ns, Width = 8 ns

ADC Value vs. Sample # (4 ns/sample)

fADC250 @ 250 MSPS
400 KHz Square Waveform on CH 1

ADC Value vs. Sample # (4 ns/sample)
JLAB Pipeline TDC

FPGA

TDC ASIC (8 channels)

Trigger

Hits

Time -->

old hits, will be erased from HIT FIFO

matching hits, will be transferred to OUTPUT FIFO

fixed trigger latency

event

time
CODA3 - Requirements/Goals

- Pipelined Electronics (FADC, TDC)
  - Dead-timeless system
  - Replacement for obsolete electronics
  - Eliminate huge numbers of delay cables
- Integrated L1/L2 Trigger and Trigger Distribution System
  - Support up to 200 KHz L1 Trigger (5µs)
  - Use FADC for L1 trigger input
  - Support 100+ crates
- Parallel/ Staged Event Building
  - Handle 100s of input data streams
  - Scalable (>1 GByte/s) aggregate data throughput
- L3 Online Farm
  - Online (up to x10) reduction in data to permanent storage
- Storage Management
  - Ordering(sorting of built events (at 15-20 kHz, 100 MB/s) to disk
CODA 3 DAQ System
Thank you!

that’s all folks...