LA-UR-08-06907

Approved for public release;
distribution is unlimited.

Title:

Data Acquisition and Instrument Control at the Lujan Center: A Retrospective

Author(s):

P. Lewis, G. Cooper, F. Trouw, D. Barr, and A. Shapiro

Intended for:

NOBUGS 2008 Conference
New Opportunities for Better User Group Software
November 3–5, 2008
Australian Nuclear Science and Technology Organisation
Sydney, Australia

LA-UR-08-06907

NOBUGS2008 Conference
New Opportunities for Better User Group Software
November 3–5, 2008
Australian Nuclear Science and Technology Organisation
Sydney, Australia

Data Acquisition and Instrument Control at the Lujan Center: A Retrospective

P. Lewis, G. Cooper, F. Trouw, D. Barr, and A. Shapiro

Manuel Lujan Jr. Neutron Scattering Center at the Los Alamos Neutron Science Center,
Los Alamos National Laboratory, Los Alamos, New Mexico USA

Abstract

The Lujan Center at the Los Alamos National Laboratory's Neutron Science Center (LANSCE) is a spallation neutron source where research in materials and biological sciences is conducted on time-of-flight neutron scattering spectrometers on twelve beam lines. Execution of an experiment on a Lujan Center neutron spectrometer involves 1) operation of the beam line components, 2) control of the sample environment equipment, 3) measurement of the neutrons scattered from the sample, and 4) presentation of the data in a form suitable for analysis. Automation and coordination of these functions is essential to effective conduct of experiments. The present data acquisition (DAQ) and control architecture was developed in the late 1990s and put into production in 2001. It has evolved over the past seven years to provide a flexible framework for automating neutron scattering experiments. However, both experimental needs and technological capabilities have also advanced significantly over the past decade, particularly in the areas of computing power and software systems. In response to this, we are re-evaluating our present systems and architecture to identify where new approaches and/or upgrades would have the most leverage.

In our present systems, the DAQ component is based on a distributed hardware and software architecture. Neutron data are collected, time-stamped, and buffered in custom VXI FPGA-based modules. Initial data processing is performed locally by VXI crate processors running VxWorks. The crate processors communicate with a Windows-based host PC over a private network. The DAQ software for both the crate and host processors was developed using high level tools based on the Executable Universal Modeling Language (xUML). A Perl API supports scripted automation of DAQ operations. Labview is generally used for controlling sample environment equipment. Individual Labview-based devices are controlled using a Labview to EPICS (Experimental Physics and Industrial Control System) interface. Both Labview and native EPICS software is employed to control beam line components. A second Perl API supports scripted automation of EPICS operations. Integrated run automation is achieved via Perl scripts, driven either from a data table or a web interface. Run data and parameters are saved in the Nexus data format. Remote access to instrument control and data acquisition, both on and off site, is achieved via direct access to the host PC using commercial virtual network connection (VNC) software.

In this paper we will review our experience with the various components and analyze and generalize the lessons learned. To summarize, our overall experience with our distributed acquisition and control architecture has been positive. On the hardware side, our custom VXI-based modules still meet our present and anticipated needs. Upgrades of the commercial VXI crate processors have allowed us to take advantage of advances in computing hardware. Our experience with the DAQ operating systems---Windows and VxWorks---has been less positive. While these were valid choices ten years ago, the dramatic growth and acceptance of Linux, along with its open source nature, make it a better choice for software upgrades. Our experience using model-based xUML tools for data acquisition software design and code generation has been disappointing. The resulting software suffered significant reliability, performance, and maintainability problems. It was not the right tool set for the application and we are phasing out its use. On the other hand, our approach to run automation via scripting interfaces has been very successful and has provided sufficient flexibility to accommodate the needs of all of our spectrometers. Our experience with the use of Labview and EPICS for slow controls software has been mixed. Labview permits quick development of equipment interfaces by a wide variety of personnel, but has significant drawback is the areas of stability, configuration control, and integration with other systems. EPICS has proven more stable and robust, but, until recently, was more complex to develop and required more expensive hardware platforms. With the advent of improved EPICS tools and platforms (e.g. soft IOC) we have begun migrating our more basic equipment control from Labview to EPICS.

Data Acquisition and Instrument Control at the Lujan Center: A Retrospective

Paul S. Lewis

Deputy Director for Operations

Lujan Neutron Scattering Center at LANSCE

Los Alamos National Laboratory

Coauthors

G. Cooper, F. Trouw, D. Barr, and A. Shapiro

NOBUGS 2008 Conference

November 3-5, 2008

Australian Nuclear Science and Technology Organisation (ANSTO)

Sydney, Australia

LAUR-08-06907

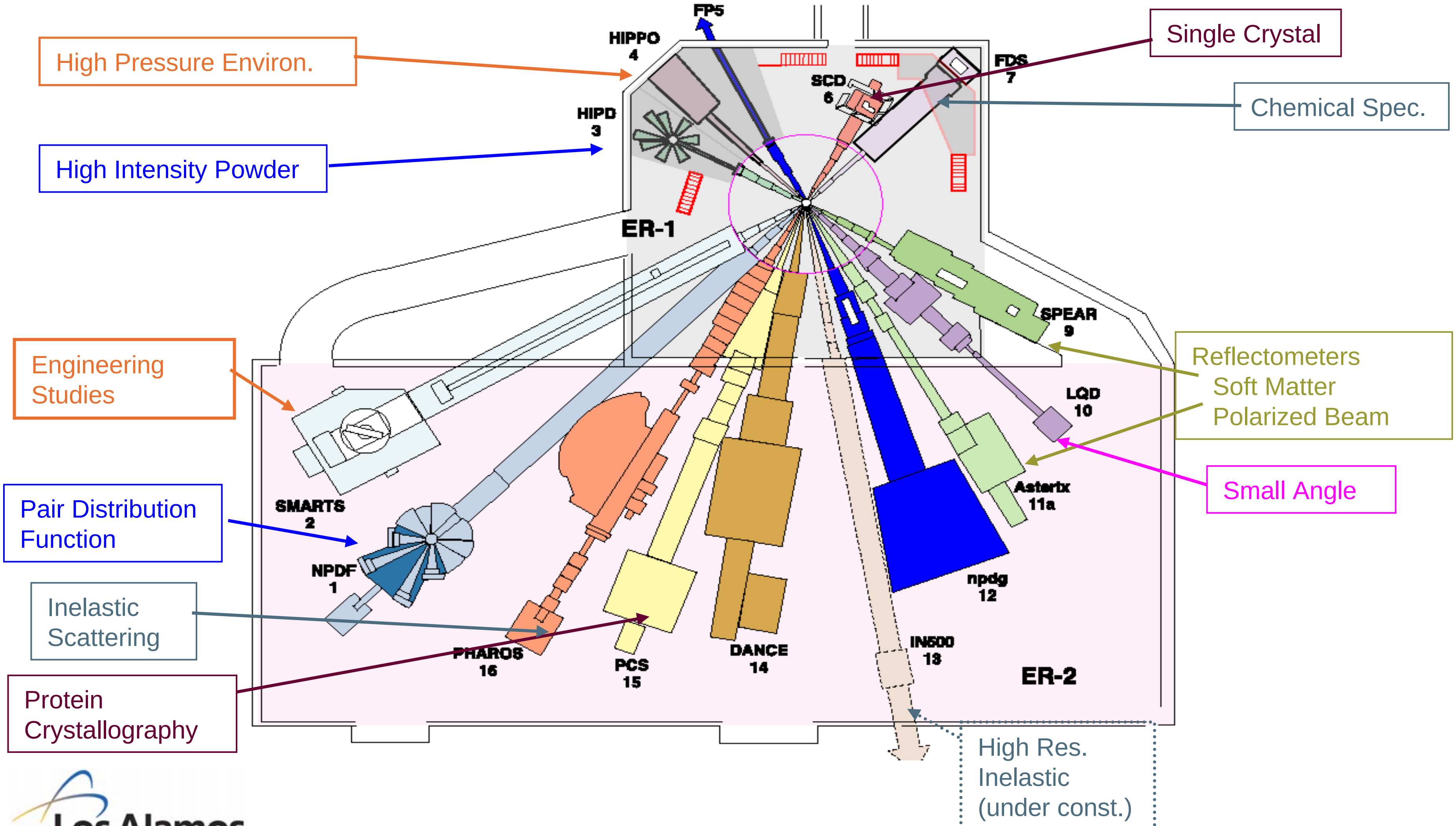
Talk Outline

- **Lujan Center Facility Overview**
- **DAQ and control history and present status**
- **Hardware architecture**
  - Description
  - Experience and retrospective
- **DAQ software**
  - Description
  - Experience and retrospective
- **Instrument control**
  - Description
  - Experience and retrospective
- **Parting thoughts**

Lujan Center is a US National User Facility for Neutron Scattering

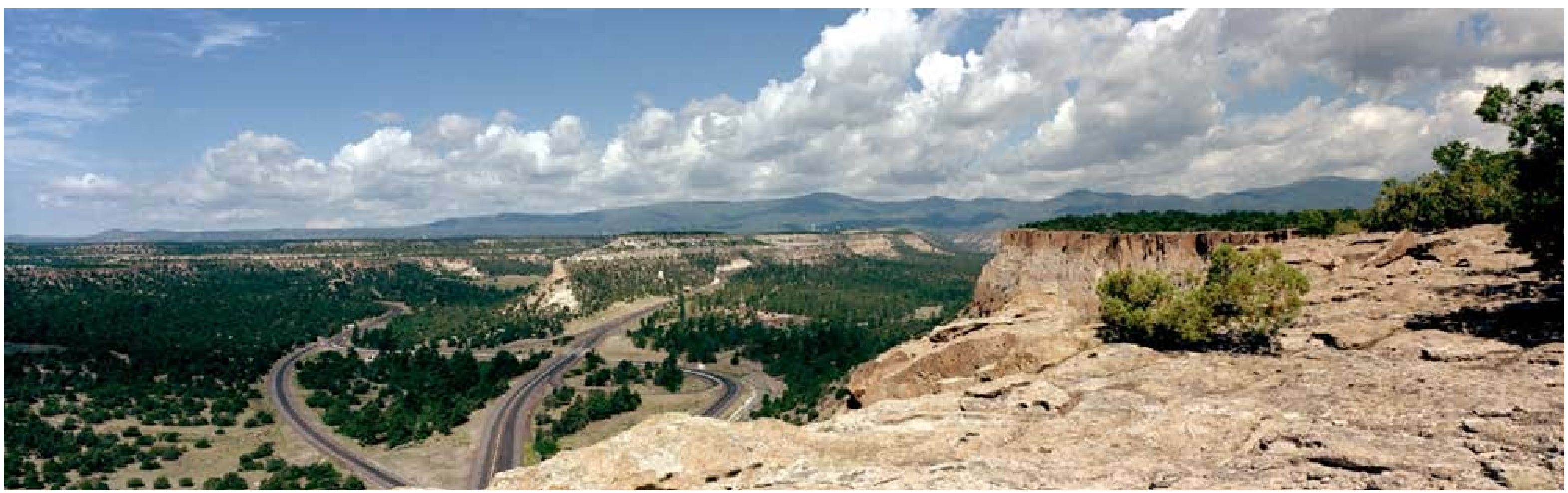
- **100 kW spallation source**
  - 800 MeV, 125 µa
  - Low rep. rate 20 Hz, high peak power
- **Tungsten target**
  - Liquid hydrogen and water moderators
- **17 beam lines** - 15 instrumented, 2 open
  - 11 operating neutron spectrometers
  - 2 nuclear physics
  - 2 development beam lines
- **US Dept. of Energy, Basic Energy Sciences, National User Program**
  - World-wide user base – universities, research labs, industry
  - Beam time awarded on a competitive, peer-reviewed basis
  - Host ~600 users per year

Neutron Spectrometers at Lujan Center

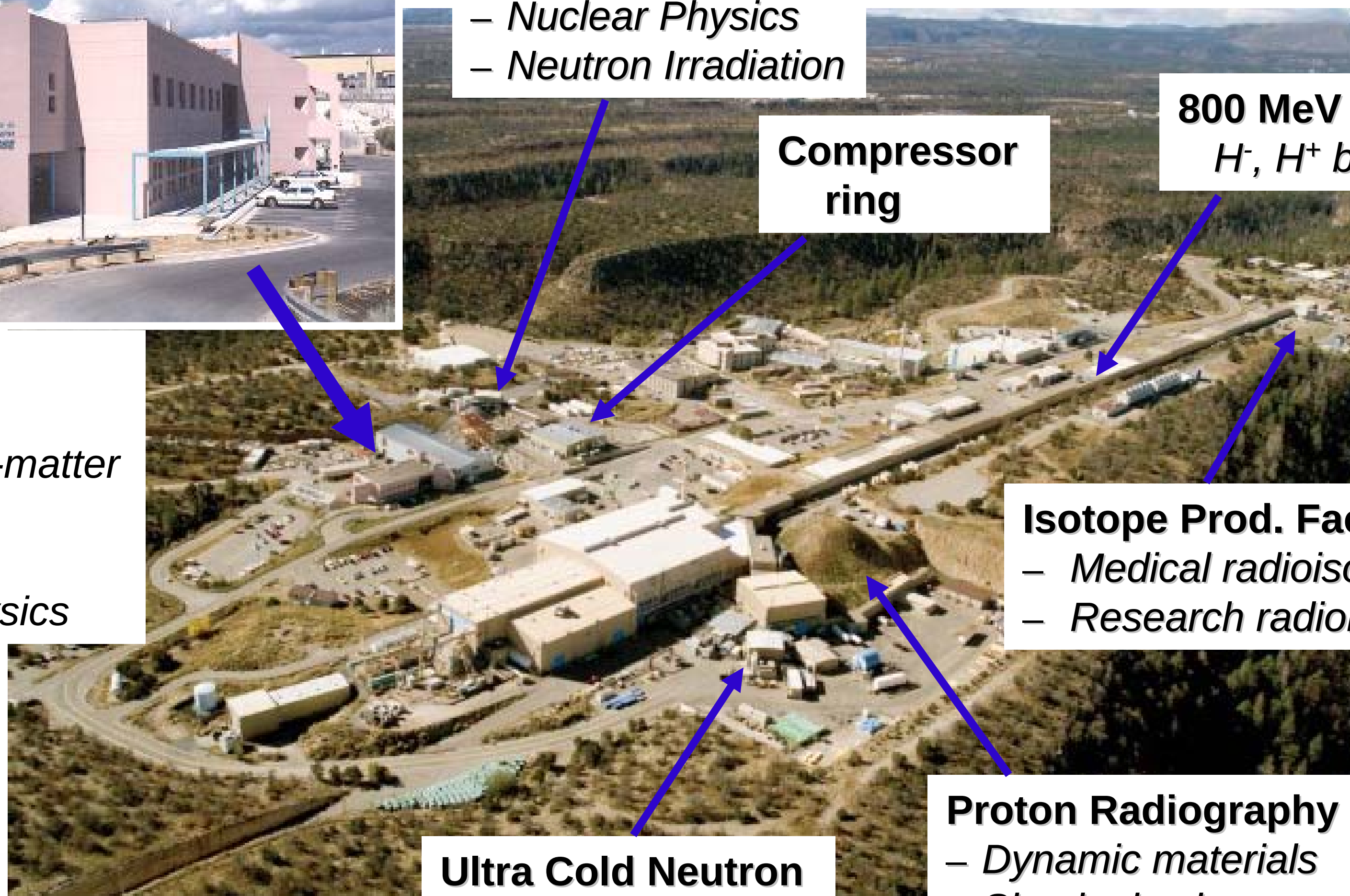


Lujan Center is a part of the Los Alamos National Laboratory

- **Los Alamos is a US Dept. of Energy national security laboratory.**
- **Located in northern New Mexico.**
- **Multidisciplinary laboratory focused on basic and applied research.**



Lujan Center is the largest experimental area at the LANSCE accelerator facility.



WNR
– *Nuclear Physics*
– *Neutron Irradiation*

Compressor ring

800 MeV linac
H^-, H^+ beams

Lujan Center
– *Materials*
– *Condensed-matter*
– *Bio-science*
– *Geophysics*
– *Nuclear physics*

Isotope Prod. Facility
– *Medical radioisotopes*
– *Research radioisotopes*

Proton Radiography
– *Dynamic materials*
– *Shock physics*

Ultra Cold Neutron
– *Basic physics*

Data Acquisition and Instrument Control System Genesis

- **SPSS Enhancement Project 1998-2002**
  - LANSCE Accelerator Upgrades
  - Lujan Center Spectrometer Development Project
- **Three new spectrometers – HIPPO, SMARTS, PCS**
- **Common Components Development**
  - Mercury Shutters
  - T0 Choppers
  - Data Acquisition – hardware and software – distributed system
- **Replaced a centralized VAX/Fastbus DAQ system**
- **Data acquisition (and instrument control) first fielded on the three SPSS instruments and Pharos – 2001-02**
- **Then implemented on all Lujan instruments – 2002-03**

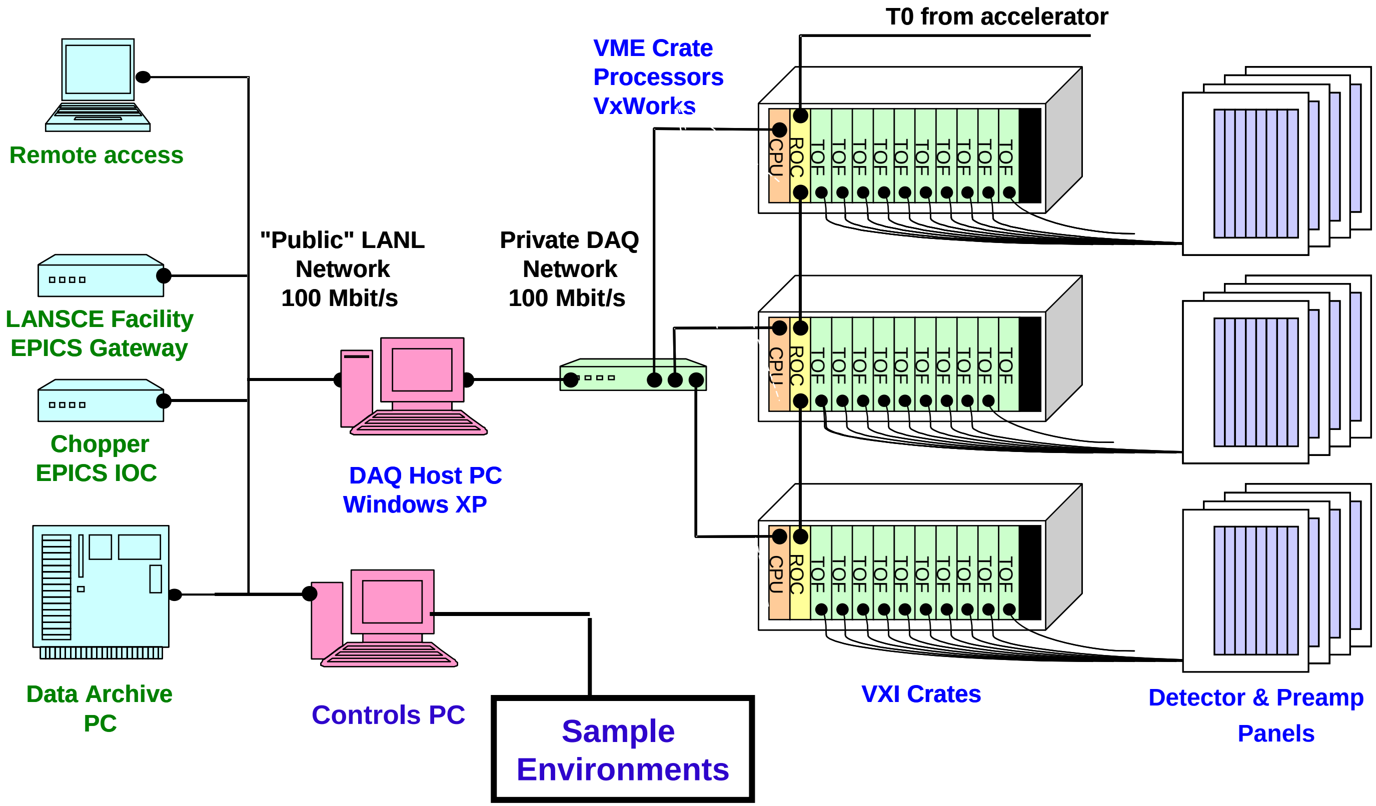
Present Data Acquisition Systems – 11 Neutron Spectrometers

Flight path	Instrument	Type	Detectors	DAQ Crates	Modules TOF / TOFA	DAQ online
1	NPDF	powder diffractometer	160 PSD 124 SED	3	29 / 0	2002
2	SMARTS	powder diffractometer	384 SED	4	25 / 0	2001
3	HIPD	powder diffractometer	128 SED	1	9 / 0	2003
4	HIPPO	powder diffractometer	1360 SED	13	101 / 0	2001
6	SCD	single-crystal diffractometer	2D multiwire area	1	1 / 1	2003
7	FDS	inelastic filter spectrometer	60 SED	1	5 / 0	2002
9	SPEAR	small angle spectrometer	1D area PSD	1	1 / 1	2002
10	LQD	small angle spectrometer	2D multiwire area	1	1 / 1	2003
11	ASTERIX	polarized beam spectrometer	1D area PSD 16 PSD	1	4 / 1	2002
15	PCS	single-crystal diffractometer	8 segment 2D multiwire area	4	1 / 8	2001
16	PHAROS	inelastic chopper spectrometer	392 PSD	5	53 / 0	2002

~ $2M (US) hardware replacement cost (not including detectors)

Hardware Architecture

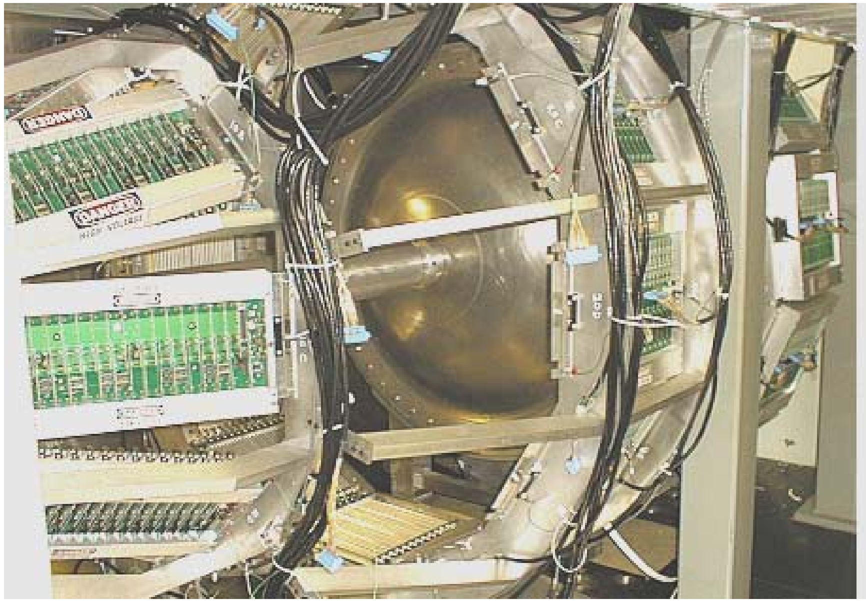
- Distributed system located at the instruments
- VXI-based DAQ hardware
- Local crate processors
- DAQ and Controls PCs



Detector Panels

- **3He detector tubes**
  - single ended or PSD
- **Motherboard-based**
- **Accommodate plug-in or wired-in tubes**
- **On-board plug-in preamps**
  - based on KEK design
  - differential output
- **On-board high voltage**

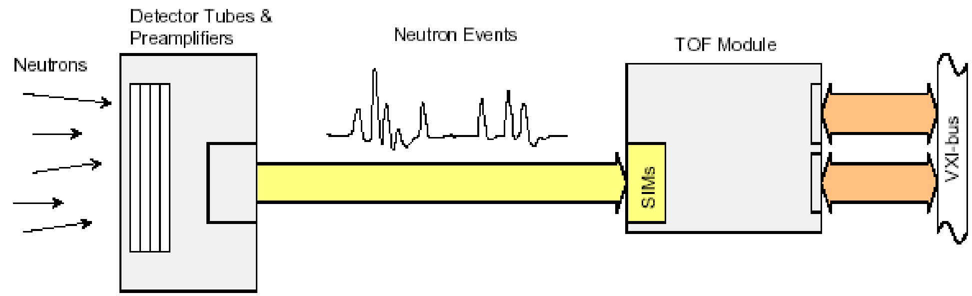
SMARTS panels detector side



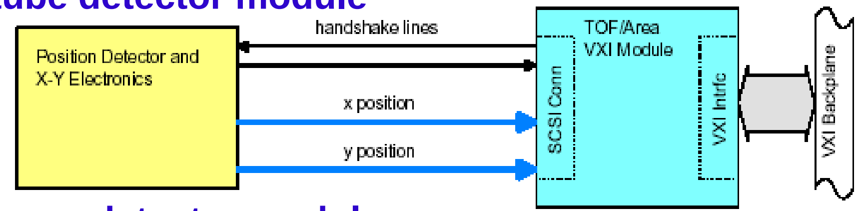
HIPPO panels preamp side

Custom Time-of-Flight Modules

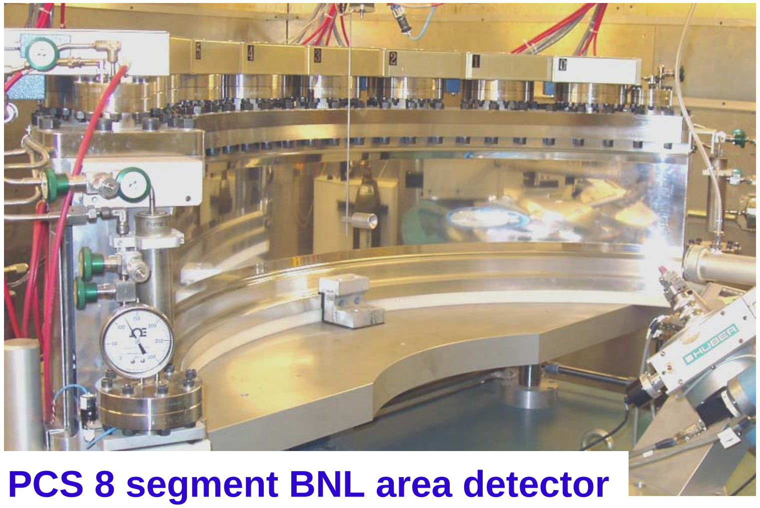
- **Custom VXI ‘C’ modules**
  - 10 MHz 32 bit bus
  - 100 ns time stamp resolution
  - FPGA based
- **TOF tube detector module**
  - Developed in collaboration with IPNS at Argonne
  - Single-ended and PSDs
  - 2K event FIFOs – odd/even
  - Instantaneous rate 330K events/sec
  - Average rate (at 20 hz) 40K events/sec
- **TOF/A area detector module**
  - 16K event FIFOs – odd/even
  - Instantaneous rate 450K events/sec
  - Average rate (at 20 Hz) 320K events/sec



tube detector module



area detector module



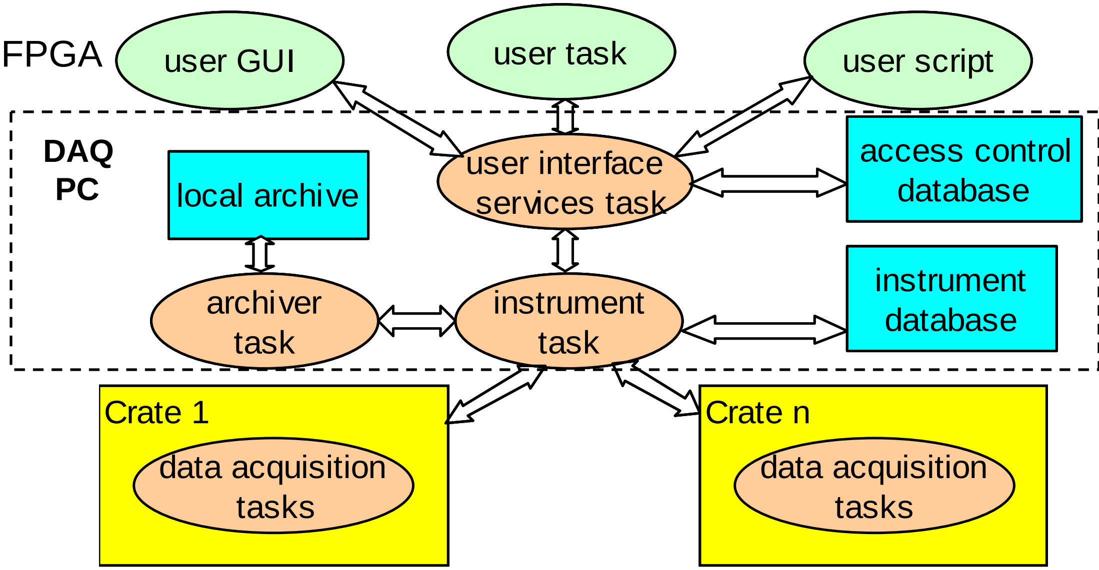
PCS 8 segment BNL area detector

Hardware Retrospective

- **Overall very positive. Reliability has been good.**
- **VXI/VME still viable and well-supported.**
- **Motherboard-based detector panels have worked well.**
  - Potential improvement: Plug in preamps – higher gain, lower noise
- **Custom TOF/TOFA modules still meet present & anticipated needs**
  - Hardware data rates still in excess of what can be supported by crate CPU hardware/software.
  - Potential improvement via FPGA reprogramming exists, but has not been utilized to date.
  - Long term concern is component availability. Most recent batch of modules fabricated in 2005. Have inherited ANL modules after IPNS closure.
- **Use of commercial crate CPUs allows us to take advantages of advances in computing hardware and operating systems.**
  - Motorola PowerPC *(400 MHz, 128Mb memory, 100 Mb ethernet)* to Intel Core Duo *(1.66 GHz, 4 Gb memory, 1 Gb ethernet)*
  - VxWorks to Linux

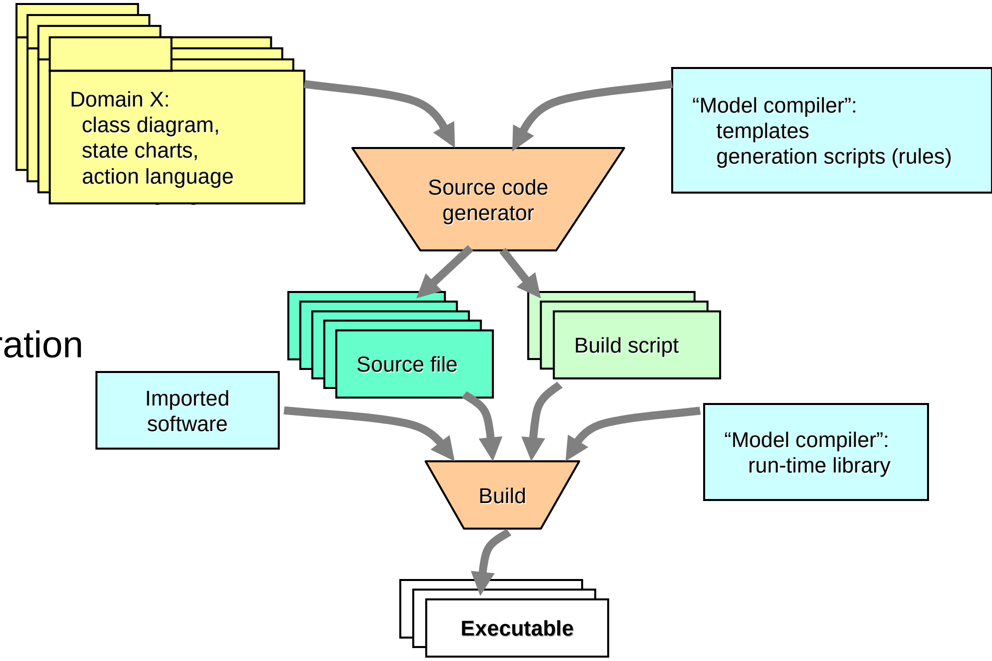
Software Architecture – Data Acquisition

- **DAQ PC – Windows NT/2000/XP**
  - User interface – local or remote
  - Acquisition run control
  - Configuration database – Access
  - Create single Nexus data files
  - Code repository: crate CPU & TOF-FPGA
- **Crate CPUs – VxWorks**
  - TOF-FPGA initialization
  - Read out TOF module data
  - Accumulate local histograms
  - 20 Hz interrupt-driven cycle
- **DAQ (UIS) APIs for Perl and Java to support run automation**



DAQ Software Development Methodology

- **Object-oriented model-based approach**
  - Project Technology tools based on Shlaer-Mellor or xUML design methodology
  - Software functionality defined as high level state-based model.
  - Model compiler generates run code and configuration database template
  - Integrated development of software across host & crate CPUs
  - Hooks available to incorporate lower level software
- **Motivations**
  - Common software automatically customized to individual instrument configuration
  - Common environment for host and crate software development
    (Intel/Windows, Motorola/VxWorks)
  - Reduce design and implementation effort.
  - Improve maintainability and documentation.



DAQ Software Retrospective – Operating Systems

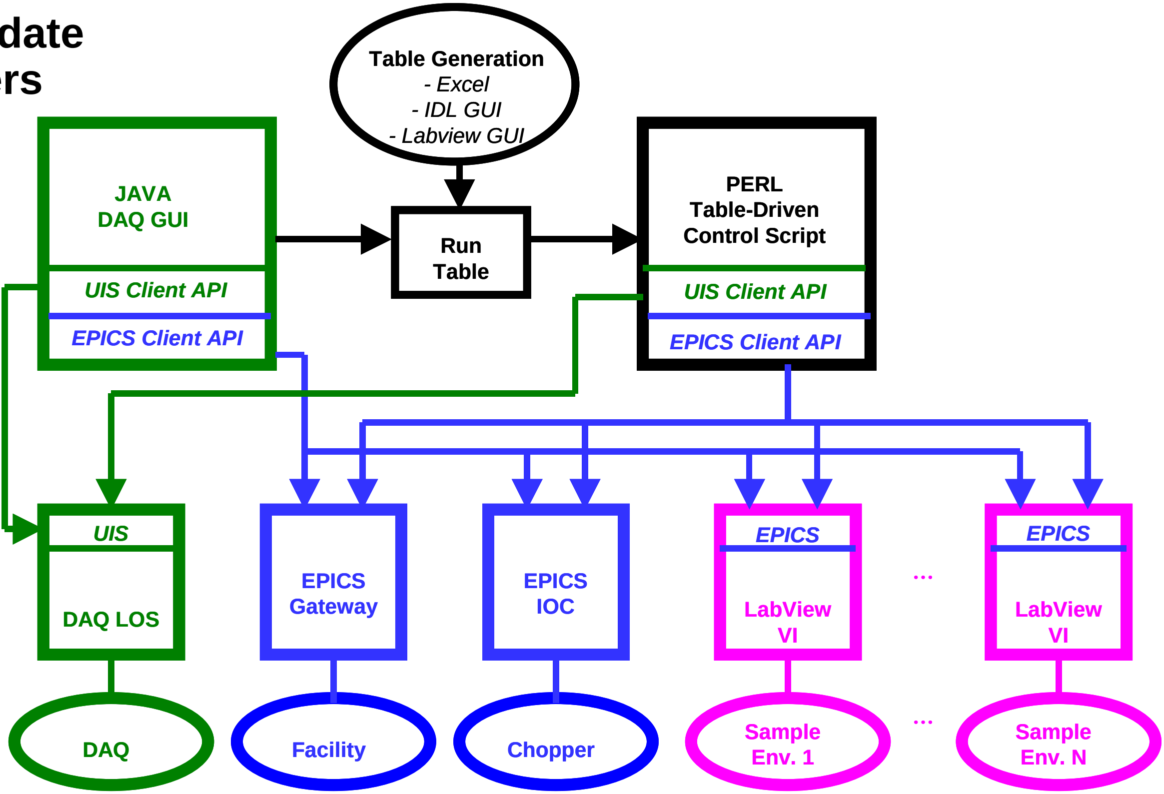
- **Windows / VxWorks was a reasonable choice in 1998**
  - Driving concern was long term viability - Unix/Linux future unclear at the time.
- **But,**
  - Persistent problems due to proprietary nature of both systems.
  - Undocumented “features” and bugs in lower level workings – particularly in TCP/IP and inter process communication
  - Windows machines “attract” other applications which complicate the operating environment.
  - Windows does not provide adequate security support for a multi-user environment.
- **Linux is now clearly a better choice**
  - Widely supported open architecture
  - Common OS on host and crate CPUs
  - More robust and flexible multi-user security
- **Our next generation DAQ software is Linux-based**

DAQ Software Retrospective – Design Methodology

- **Our experience with the Shlaer-Mellor/xUML based design tools has been unsatisfactory.**
  - DAQ software has been unreliable, hard to maintain, and difficult to upgrade.
  - Placed significant restrictions on functionality, e.g. real time data viewing
- **Object-oriented model-based tools not up to the task.**
  - Long start up and learning curve
  - Significant amounts of hard-coded software required to fill in the gaps for a real time application of this sort.
  - Significant problems with invisible “inner working” of the model produced code, particularly in inter process communications.
  - Very difficult to incorporate hardware or software upgrades for crate CPUs
  - Model-produced code driven by a signal common database with ill-defined data dependencies.
  - We spent considerable effort attempting to resolve problems, but ultimately found them to be endemic.
- **We have abandoned these tools for our next generation DAQ software**

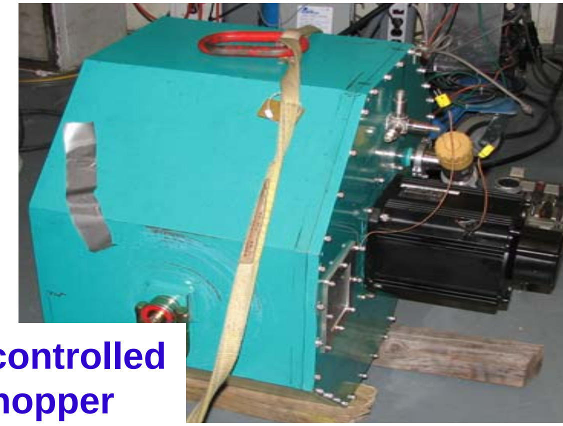
Instrument Run Automation

- **Automate and coordinate all aspects of instrument operation**
  - DAQ data collection
  - Sample environment operation – temperature, pressure, etc
  - Beam line components – shutters, choppers, collimators, etc.
  - Error recovery
- **Open architecture to accommodate a wide variety of local controllers**
  - Labview devices
  - EPICs devices
- **Perl script-based table-driven control for flexibility**
- **Supports range of table generation approaches**
  - Excel spreadsheets
  - IDL / Labview based GUIs
  - Web-based generators



EPICS-Based Control Interface

- *Experimental Physics and Industrial Control System*
- **EPICS-based facility interface**
  - Moderator temperature, beam current, etc.
- **EPICS-based beamline components**
  - Shutters, choppers, etc
- **EPICS-enabled LabView control of sample environments.**
  - Locally-written LabView extension to “publish” LabView variables in EPICS domain.
  - Allow “non-programmer” development of control software.
- **EPICS APIs for Perl and Java to support control automation.**



EPICS-controlled T0 chopper

LabView-controlled load frame

Control Retrospective

- **Script-based approach has been high successful**
  - Flexibility to accommodate wide variety of instrument needs
  - Leverage instrument teams to develop user interfaces to control tables
  - Perl APIs for EPICS and DAQ have been robust
- **Labview control of sample environments has been mixed**
  - Relatively easy to develop – leverage instrument teams
  - Labview VIs and interfaces readily available for commercial hardware
  - Software difficult to maintain and very machine/installation dependant
  - Licenses/upgrades expensive
  - No native means to network - EPICS interface locally written and unsupported
- **EPICS – almost the mirror image**
  - Has been much more complex to implement – professional programmers (Changing now with the availability of softIOC support)
  - Not a commercial standard (but widely used at scientific facilities)
  - Robust, mature, free – good local support
  - Integral distributed/network support
- **Will continue to live with both, but plan to move the more basic and widespread controllers (e.g. temperature controllers) to EPICS softIOC on Linux.**

Parting Thoughts

- **An integrated control environment is critical**
  - Command-based interfaces to all components to support flexible central control
  - Remote control/access must cover all aspects of instrument operation
- **DAQ systems can be reasonably homogeneous across instruments and stable over time**
  - Single DAQ system hardware/software accommodates all – one size fits all
  - Common support team
- **Sample environments and beam line components are the opposite – vary widely from instrument to instruments and change from experiment to experiment**
  - One size does not fit all – greater customization and flexibility required.
  - Need an adaptable open architecture that can evolve with demands.
  - Must depend on instrument personnel for much of the automation
  - Need good interface tools usable by non-programmers
- **TCP/IP and inter-process communication can be a point of great weakness, particularly for “fast” data or control.**
- **Resources to support and improve/upgrade systems are much more difficult to come by than the resources for initial development – plan accordingly**