

Beyond ACNET:

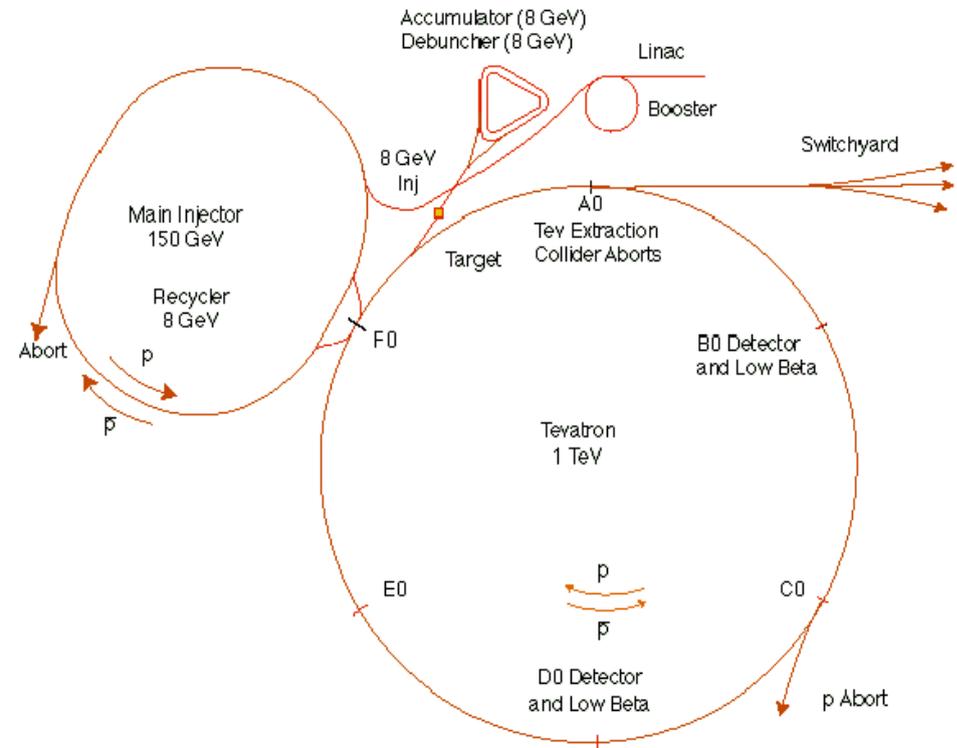
Evolution of Accelerator Control System at Fermilab

Andrey Petrov,
Fermi National Accelerator Laboratory
Accelerator Controls Department

Presentation at SLAC • March 17, 2009

The Origin of ACNET

- ACNET usually refers to the Accelerator Control System at Fermilab.
- The system was developed in early 1980's for the Tevatron.
 - Still based on the original architectural design.
 - The implementation has changed over time.
- Currently serves the entire Accelerator Complex.
 - Supports multiple operation modes simultaneously.
- Will likely stay in the future.



Fermilab Accelerator Complex

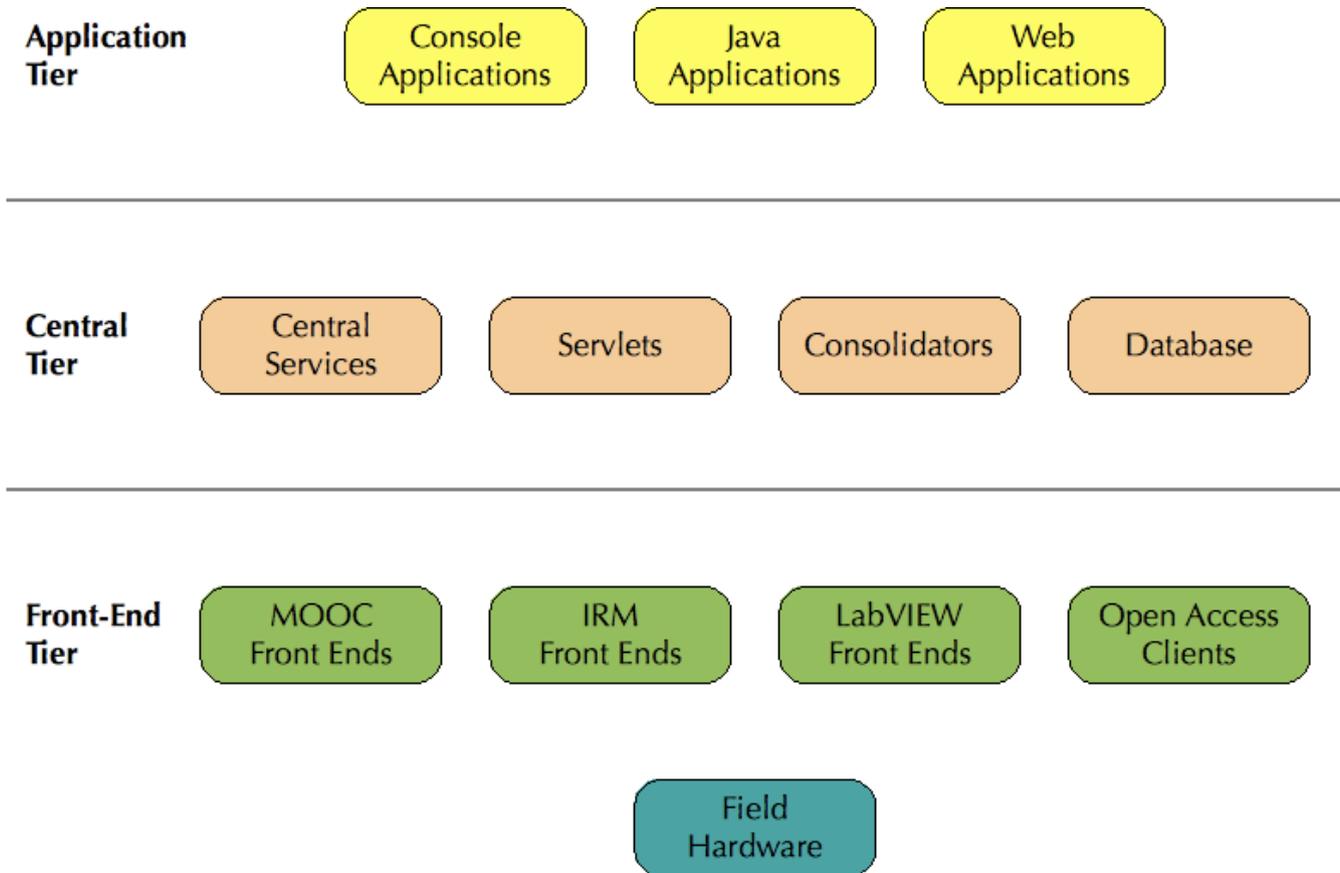
ACNET = ACcelerator NETwork

- In fact, **ACNET** is a suite of common application and transport protocols used in the Control System.
- Designed for effective real-time data acquisition.
- Example protocols:
 - **RETDAT/SETDAT** — RETurn DATa/SET DATa.
 - **Get32/Set32** — an extension of RETDAT/SETDAT.
 - **FTP** — Fast Time Plot, groups of readings @ ≤ 1.44 kHz.
 - **Snapshot** — Blocks of 2048 points.
- Based on:
 - DEC's Parallel Communication Link (PCL) — 1980's.
 - IEEE 802.5 token ring.
 - Ethernet, carried on top of UDP — 2000's.
- Currently implemented on Linux, VxWorks, and in Java.

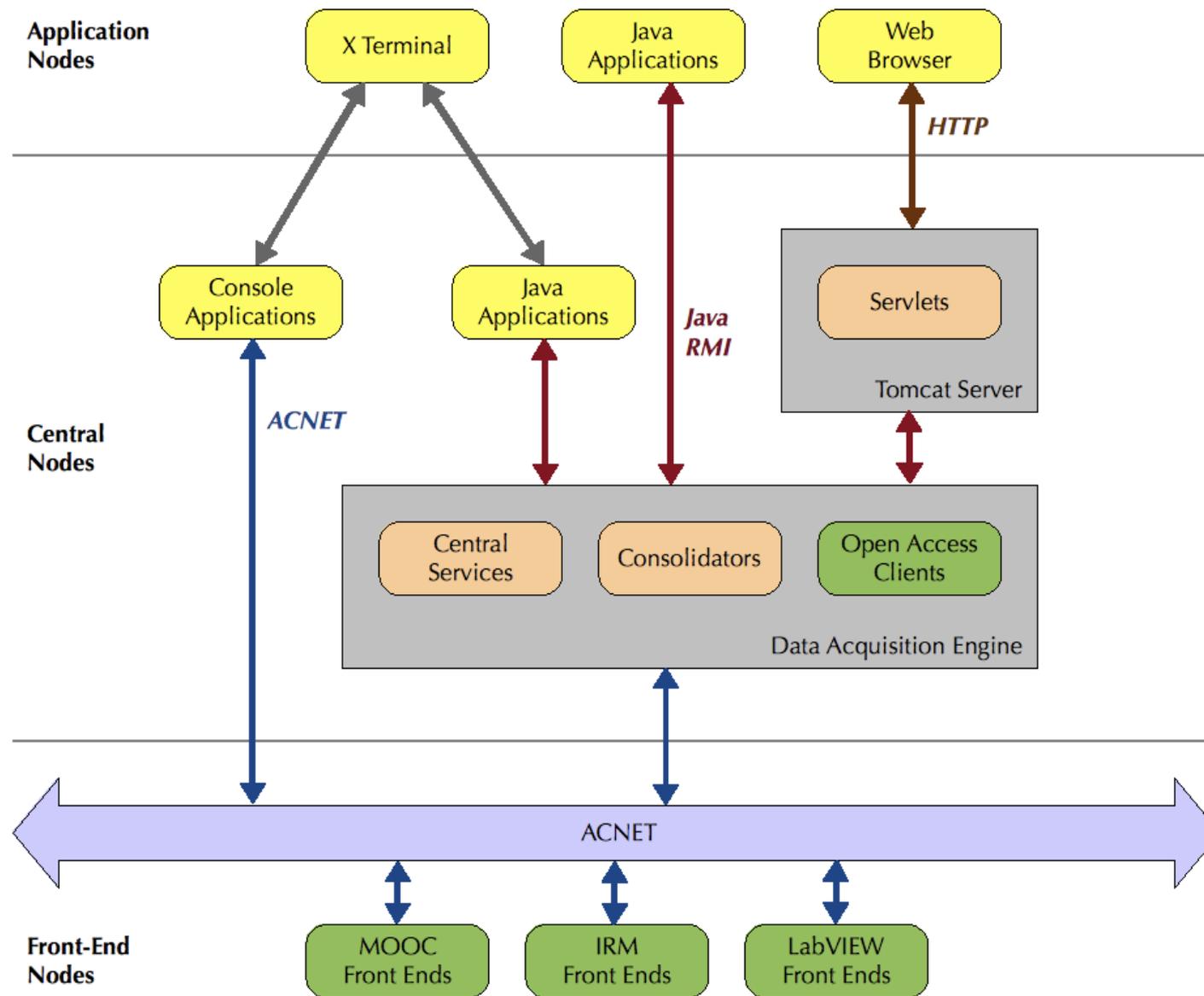
Device Model

- A flat namespace, each device name is ≤ 8 chars.
 - e. g., in **T:ERING** (*Tevatron Ring Energy*) **T** means the Tevatron and **ERING** designates a device within that subsystem.
- Each device may have up to 6 properties:
 - **READING**, **SETTING**, **BASIC_STATUS**, **BASIC_CONTROL**, **ANALOG_ALARM**, and **DIGITAL_ALARM**.
- Properties can be readable, writable, or both.
- A property value can be expressed in 3 scales:
 - **RAW**, **PRIMARY**, and **COMMON**.
- Each kind of the properties uses a pre-defined data type or an array of that type.
- Currently there are ~200,000 devices and ~350,000 properties.
- Recently implemented long (≤ 64 chars) devices names.

Logical Organization



Physical Organization



Front-Ends

- Acquire data from the hardware and field buses in response to timing signals. Provide mapping between the central device database and the actual hardware readings and settings.
- Use ACNET protocols to talk to the rest of the control system.
- 3 base front-end architectures:
 - **MOOC** (*Minimally Object-Oriented Controls*). ~275
VME/VxWorks. Used in the Tevatron, Main Injector, and Pbar Src.
 - **IRM** (*Internet Rack Monitor*) & **HRM** (*Hotlink Rack Monitor*)
VME/pSOS/VxWorks. Used mostly in Linac. ~185
 - **OAC** (*Open Access Clients, a.k.a. Software Front Ends*). 131
Java-based. Run on central nodes.
- Also there are ~25 LabVIEW front ends, mostly in various instrumentation subsystems.

Data Acquisition Engine (DAE)

- A custom Java application server. 123
- Provides:
 - Data consolidation.
 - A front-end façade for applications.
 - Central services.
 - Open Access Clients (software front-ends).
- Talks ACNET protocols to the front-ends.
- Talks Java RMI to the applications.
- Runs on uniformly configured nodes.
 - Now Solaris and Scientific Linux.
 - Solaris will be replaced with Linux soon.
 - Co-located with Tomcat and MySQL (for data loggers).

Central Services

- Data Loggers. ~70
 - Each data logger: 60 devices, $5 \cdot 10^8$ total points, (1446 days @ 15 seconds). Local MySQL tables.
- Sequenced Data Acquisition (SDA).
 - A complex data logger saving data selectively, depending on the machine state.
- Save/Restore.
 - Automatically saves the system's state 4 times a day.
- Alarms.
- Front-End Download.
- Accountability.
 - Logs all settings done in the control system.

(Legacy) Console Application Environment

- ~600 applications written in C++.
 - Each application is usually for a single purpose.
 - The framework is simple and robust.
 - ※ Yet, considered outdated by many – especially the GUI.
 - This is currently the main tool to operate the accelerators.
- May run only on central nodes, currently Linux. ~70
 - Access via X terminal.
 - Access via a lightweight read-only Java client.
- 1 client = 3 text pages + 3 (9?) graphical screens.

Example of a Console Application

The screenshot shows a Java console window titled "PA F4 COMPRESSOR PARAMS<NoSets> - ACNET Java Console". The window has a menu bar (File, View, Help) and a toolbar with buttons for PA, PB, PC, SA, SB, SC, Util, GxPA1, and GxPA2. The main display area shows a terminal-style interface with the following content:

```

F4  A0 COMPRESSOR SURVEY          SET      D/A  A/D  Com-U  ♦PTools♦
-<FTP>+ *SA♦ X-A/D  X=TIME          Y=E:DIAQR1,9:DIAQWI,E:DIAQWO,E:DIAQWD
COMMAND ... Eng-U  I= 0          I= 0      , 0      , 0      , 0
-<A0>+  Once AUTO  F= 300        F= 2000   , 2000   , 2000   , 2000
SURVEY comp_1 comp_2 comp_3 comp_4  invntry miscell valvpos

T:A0PI1      SUCTION PRESSURE                1.28    PSIG
T:A0PI2      DISCHARGE PRESSURE              290.2   PSIG
T:A0FI2M     Bldg GHe FI2 CalcFlow                   176.2   g/s
T:A0FIBM     EVHP Bypass FIBY CalcFlw                 0       g/s
-T:A0EVHP    COMPR BLDG BYPASS VALVE            0        0       %OPE  ..

T:A0TEMP     I/O CRATE ADC CARD TEMP                   99.66   DEGF

-T:A0HV1L    COMP#1 LS SLIDER POS                       100     96.77   %LOA  ..
-T:A0HV2L    COMP#2 LS SLIDER POS                       100     99.61   %LOA  ..
-T:A0HV3L    COMP#3 LS SLIDER POS                       100     * 22.48 %LOA  ..
-T:A0HV4L    COMP#4 LS SLIDER POS                       100     99.8    %LOA  ..
T:A0Pw30     A0 #1 COMPRESSOR POWER                    274.3   KW
T:A0Pw40     A0 #2 COMPRESSOR POWER                    229.8   KW
T:A0Pw50     A0 #3 COMPRESSOR POWER                    *-99.99 KW
T:A0Pw60     A0 #4 COMPRESSOR POWER                    265.1   KW

T:A0CTN2     CONTAMINATION N2/HE                        -.147   PPM
T:A0CTWR     CONTAMIN H2O/HE 58742                     2.184   MH

T:A0PIWS     Water Supply Pressure                     129     PSIG
T:A0WPST     COOLING WATER PUMP START                  .
T:A0FUPS     A0 FRIG UPS ALARMING                      .
T:A0VTPS     TRANDUCR PWR SUP VOLTAGE                  22.77   VOLT
  
```

At the bottom of the console, a status bar indicates "Connected to clxsr.v.fnl.gov:1920".

Java Application Environment

- ~ 130 applications.
 - Again, each application is usually for a single purpose.
 - There is a custom application framework that facilitates development of GUI and implementation of principal functions.
 - ※ Yet, considered too complicated and not quite functional by many.
 - Used not as widely as the legacy environment.
- May run on client machines, as well as on central nodes.
 - Access via X terminal, if on a central node.
- Applications are launched by Java Web Start.
 - The main complaint – too slow.

Example of a Java Application

The screenshot shows a Java application window titled "F4 - JPPage" with a menu bar (File, Edit, Search, View, Data, Tools, Help) and a toolbar. The main content area displays a table of compressor parameters. On the left, a tree view shows the hierarchy: F4 + COMPRESSOR PARAMS > SURVEY > A0 + A0 COMPRESSOR. The table has columns for parameter names, descriptions, and values with units. The table is titled "F4. A0 COMPRESSOR SURVEY" and shows data as of "15-Mar-2009 22:05:12".

		SET	D/A	A/D	
T:A0PI1	SUCTION PRESSURE			1.221	PSIG
T:A0PI2	DISCHARGE PRESSURE			286.8	PSIG
T:A0FI2M	Bldg GHe FI2 CalcFlow				
T:A0FIBM	EVHP Bypass FIBY CalcFlw				
T:A0EVHP	COMPR BLDG BYPASS VALVE		0	0	%OPE **
T:A0TEMP	I/O CRATE ADC CARD TEMP			101.6	DEGF
T:A0HV1L	COMP#1 LS SLIDER POS		100	96.77	%LOA **
T:A0HV2L	COMP#2 LS SLIDER POS	72.13	74.78	73.02	%LOA **
T:A0HV3L	COMP#3 LS SLIDER POS		100	100	%LOA **
T:A0HV4L	COMP#4 LS SLIDER POS		100	99.8	%LOA **
T:A0PW30	A0 #1 COMPRESSOR POWER			271.4	KW
T:A0PW40	A0 #2 COMPRESSOR POWER			152.6	KW
T:A0PW50	A0 #3 COMPRESSOR POWER			-99.99	KW
T:A0PW60	A0 #4 COMPRESSOR POWER			267.6	KW
T:A0CTN2	CONTAMINATION N2/HE			1.172	PPM
T:A0CTWR	CONTAMIN H2O/HE 58742			2.203	MH
T:A0PIWS	Water Supply Pressure			131.4	PSIG
T:A0WPST	COOLING WATER PUMP START				*
T:A0FUPS	A0 FRIG UPS ALARMING				*
T:A0VTPS	TRANUDCR PWR SUP VOLTAGE			22.76	VOLT

At the bottom of the window, there are three status boxes: "COMMON UNITS", "Read Only", and "Source: Servlet".

Web Applications

- ~ 50 applications.
- Organized as servlets running in Tomcat containers.
- Serve a variety of purposes:
 - Convenient way for the users to access data.
 - ※ e.g., Synoptic, SDA Viewer, Supertable.
 - Lightweight protocol for client programs.
 - ※ XML-RPC and REST.
 - Viewing and editing data from the database.
 - ※ Java Application Index, ACNET Device Database (D80).
- Kerberos authentication in the browser via a custom applet.

Console Apps. vs. Java Apps.

- CVS Code Repository.
- Command-line building system (MECCA).
- Applications usually acquire data via ACNET protocol.
- Applications always run on central nodes.
- Applications are started from Index Pages.
 - Now capable to start Java applications, EDM screens.
- CVS Code Repository.
- A Web-based building system.
- Applications usually acquire data from DAE via Java RMI.
- Applications usually run on client nodes.
- Applications are usually started from the web.

An Index Page

The screenshot shows a Java console window titled "PA:D <INDEX> Class: <WebUser> - ACNET Java Console". The window has a menu bar with "File", "View", and "Help". Below the menu bar are several tabs: "PA", "PB", "PC", "SA", "SB", "SC", "Util", "GxPA1", and "GxPA2". The main content area displays a "Diagnostic & Utility Index Page" with a list of items organized in three columns. The items are numbered 1 through 69. Some items are highlighted in yellow or green. At the bottom of the main content area, there is a "Messages" section with a search bar and navigation arrows. The status bar at the bottom indicates "Connected to clxsr.v.fnal.gov:1920".

Diagnostic & Utility Index Page			◆ Cmnds ◆◆ Pgm_Tools ◆
1 Save/Restore	24 BS Water & Air	47 Beam Switch Box	
2 RAD SAVE/RESTORE	25	48 FTP DIAGNOSTICS	
3 DPM Usage Peek	26	49 SNAPSHOT PLOT TEST	
4 Graphics Tests	27 Array Device Plot	50 Filesharing Dump	
5 Gx Hardcopy/Save	28 Lex Draw	51 Graphics Stats	
6 Alarm Screen Setup	29	52 Alarm Download	
7 Quick SA Test	30 Node Poll	53	
8	31 ACNET Node Poll	54 Settings History	
9 Console Manager	32	55 Device Broken/OOS	
10 Screen Image Edit	33 Clockscope	56 C190 TEST	
11	34 Plot Annotation	57 GAS ALARM DOWNLOAD	
12 SLD Statistics	35 XConsole CachePeek	58	
13	36 Generic Structures	59 Alarm/List Control	
14 REMOTE INSTRUMENT	37 Curve Fit II	60 Lock Peeker	
15 Console Peek	38 Level II Exerciser	61 TEV CLOCK PAGE	
16 Radix Conversion	39 GAS Test/Boot/Etc	62	
17 Phone Index	40 BPM DIAGNOSTICS	63 ACCELERATOR NUMBRS	
18 Downtime Logger	DATA LOGGING	64 SYNCH DESIGN PLAY	
19 Digital Status	42 Snapshot Logger	65	
20 CAMAC Link Status	43 Lumberjack Config	66 SCANNED IMAGES	
21 CAMAC Diagnostics	44 Lumberjack Plotter	67 Alarm Control	
22 CLIB Peeker	45 Array Datalogger	68 Alarm Display	
23 MYCALC - Calc Pgm	46 Lex SA Companion	69 Timeline Generator	

Application Index

The screenshot shows a web browser window titled "Application Index" with the URL <http://www-bd.fnal.gov/appix/>. The browser's address bar and search bar are visible. The website has a dark green sidebar on the left with the following menu items: "Application Index main menu", "apetrov [logout]", "[java controls]", "[home]", "[getting started]", "[documentation]", "[statistics]", "Launch: [from the web] [in application browser]", "Manage: [applications] [servers]", "Search applications: [input field] [go]", and "[security·privacy·legal]". The main content area features the Accelerator Controls Department logo and a navigation bar with "Getting Started", "Documentation", "Start Application", and "Get Java™". The main text welcomes users to APPIX, the database of Java™ controls applications, and provides instructions on how to use the site. It also lists "New Applications" and "Recently Used Applications". The "New Applications" list includes: Beam Phase Monitor, Debuncher Signal To Noise, Fast Time Plot, Garbage Collection Info, HINS Timer Plot, Linac Tune, Pbar Debuncher 53MHz BPM, Pbar Debuncher to Acc Sem806 Display, Pbar Performance Calculator, Pelletron Protection, and Resistive Wall Mon Long Emit. The "Recently Used Applications" list includes: Java Parameter Page Lite. At the bottom of the main content area, there is a "[legend]" link and a "[security·privacy·legal]" link.

Principal Applications

- **Parameter Page**

The most universal program that controls a list of devices.

- **Fast Time Plot**

Plots device data in real time @ ≤ 1.44 kHz.

- **Snapshot Plot**

Plots device data @ ≤ 20 MHz.

- **ACL** (*Accelerator Command Language*)

A data acquisition scripting tool.

- **Sequencer**

The primary program for coordinating operations of the accelerator complex. Uses ACL.

- **Lex SA** (in Console Framework) / **Synoptic** (in Java)

Synoptic graphics programs.

Status of Java at Fermilab Controls

- The platform was introduced back in ~1996. Java applications were to be the path away from VMS.
- Yet, after 10 years Java applications didn't replace the legacy framework:
 - (For many) the programmatic API is too hard.
 - ※ Obviously, programming Swing with dynamic layouts/events/etc. is very hard for an occasional programmer.
 - (For many) the development environment is too hard.
 - Development of 2-tiers applications is harder than 1-tier.
 - Startup time is perhaps the biggest annoyance for the users.
 - Java RMI–related issues: unmarshaling, local callback observers.
 - Participation from machine departments didn't happen.
 - Only new applications, which don't have a counterpart in the legacy system, get eventually adapted by the users.

Status of Java, continued

- Meanwhile, the middle tier is now implemented in Java.
 - Performance, in general, is seemed to be satisfactory.
 - ※ Now moving from Sun Netras to i386 Linux.
 - A lot of issues with erratic behavior of garbage collection.
 - Subtle changes between JVM versions may cause different behaviors of Data Acquisition Engines.

Linux Migration

- Prior to 2007 the legacy part of the control system ran on VAX/VMS platform.
- It seemed to be unlikely that ~500 existing applications would ever be rewritten in Java.
- In 2004-2006 the system was migrated to Linux.
 - All non-obsolete VMS applications were ported.
 - ※ Things like VMS mail application were left behind.
 - No changes to the existing look-and-feel.
 - No specific changes to behavior of the applications.

Emergence of EPICS

- *As a part of the International Linear Collider (ILC) R&D Program, Fermilab has constructed new facilities for testing superconducting RF technologies for a future accelerator.*
- In 2005 it was decided to use EPICS and DOOCS control systems at ILC Test Area (ILCTA), and install them independently from the main ACNET system.
 - More lightweight.
 - Better collaboration with other institutions.
 - Using existing hardware developed elsewhere.
- Over time, as ILC R&D has slowed down, ILCTA was converted for Fermilab's *Project X* (a new superconducting linac).
- An instance of the EPICS control system is currently in place and running.

Status of EPICS at Fermilab Controls

- Now the two control systems are mostly independent.
- EPICS PVs can not be accessed from ACNET applications.
- EDM screens can be started from the ACNET index pages.
- EDM screens can display both PVs and ACNET devices.
- EPICS timing signals are available in ACNET.
- EPICS will stay in the future as a subsystem.
- It will be possible to read EPICS PVs through ACNET.
 - ACNET-aware IOCs.
 - A programmatic EPICS-to-ACNET bridge.
- EPICS users will be able to take advantage of the existing central services and applications.

Future of Fermilab

- The Tevatron will eventually be shut down (so far, 10/2010).
- The complex will be upgraded to increase the intensity delivered for the neutrino program.
- New NOvA experiment is expected to run until the late 2010's.
- Existing superconducting RF test facilities:
 - HINS – *High Intensity Neutrino Source*.
 - NML – *New Muon Lab*.
- Project X – a new 8 GeV superconducting linac.

Future of ACNET

- ACNET will be used as a foundation for a new control system in the post-Tevatron era.
 - A single system for the entire accelerator complex.
- Full integration with EPICS front-ends.
- Replacement of obsolete hardware.
- Modernizing legacy and Java application environment.
- A number of issues (some are under investigation now):
 - Better GUI framework and infrastructure.
 - Better client application protocols.
 - Better data pool managers (in the middleware).
 - Operating with complex data structures, self-describing data sets.
 - ...

References

- Fermilab Accelerator Controls Department,
<http://www-bd.fnal.gov/controls>
- Accelerator Division Document Database,
<http://beamdocs.fnal.gov/AD-public/DocDB/DocumentDatabase>
- An overview of existing control systems,
<http://www-bd.fnal.gov/icfabd/Newsletter47.pdf>
- Project X Control System Requirements,
<http://beamdocs.fnal.gov/AD-public/DocDB/ShowDocument?docid=2934>