NuMI Online Beamline Monitoring using JAS

and Offline Data Analysis Results

Mary Bishai, Mark Dierckxsens,Brett Viren

mbishai@bnl.gov, bviren@bnl.gov, mdier@bnl.gov

Brookhaven National lab

BROOKHAVEN NATIONAL LABORATORY

What Is JAS? Unnecessary Jargon:

The goal of the Abstract Interfaces for Data Analysis (AIDA) project is to "define abstract interfaces for common physics analysis objects, such as histograms, ntuples, fitters, IO etc.. The adoption of these interfaces should make it easier for physicists to use different tools without having to learn new interfaces or change all of their code. Additional benefits will be interoperability of AIDA compliant applications (for example by making it possible for applications to exchange analysis objects via XML)."
http://java.freehep.org/jaida/index.html

- JAIDA is the JAVA language implementation of AIDA.
- Java Analysis Studio 3 (JAS3) is a full featured GUI which uses JAIDA internally and runs in real time.

BROOKHAVEN NATIONAL LABORATORY

JAS Examples - Histogramming

JAS3

File Edit View Tuple Run Window Help

Examples | CreateAndPlotHistograms.java

```
import java.util.Random;

public class CreateAndPlotHistograms
{
  public static void main(String[] argv)
  {
    IAnalysisFactory af = IAnalysisFactory.create();
    ITree tree = af.createTreeFactory().create();
    IHistogramFactory hf = af.createHistogramFactory(tree);

    tree.mkdir("/Histograms");
    tree.cd("/Histograms");

    IHistogram1D h1 = hf.createHistogram1D("Histogram 1D",50,-3,3);
    IHistogram2D h2 = hf.createHistogram2D("Histogram 2D",40,-3,3,40,-3,3);

    tree.mkdir("/Clouds");
    tree.cd("/Clouds");

    ICloud1D c1 = hf.createCloud1D("Cloud 1D");
    ICloud2D c2 = hf.createCloud2D("Cloud 2D");

    IPlotter plotter = af.createPlotterFactory().create("CreateAndPlotHistograms.java plot");

    plotter.show();
    plotter.createRegions(2,2);

    plotter.region(0).plot(h1);
    plotter.region(1).plot(h2);
    plotter.region(2).plot(c1);
    plotter.region(3).plot(c2);

    Random r = new Random();

    for (int i = 0; i < 100000; i++) {
      h1.fill(r.nextGaussian());
      h2.fill(r.nextGaussian(),r.nextGaussian());
      c1.fill(r.nextGaussian());
```

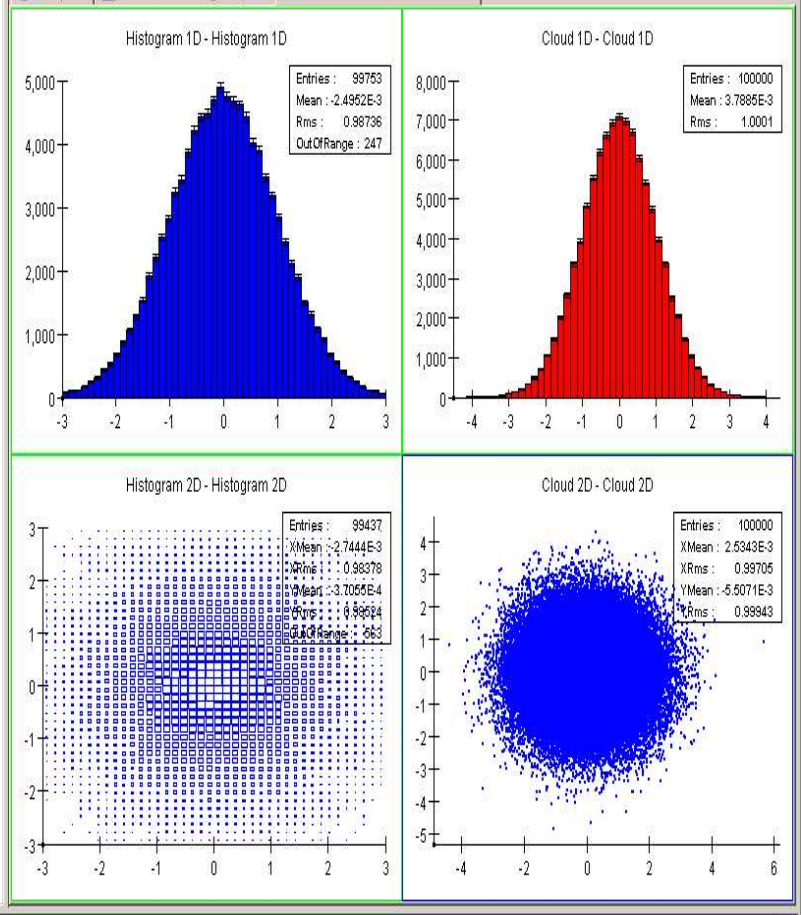
The GIMP

JAS3

File Edit View Tuple Run Window Help

Programs
tree-0

Examples | CreateAndPlotHistograms.java | CreateAndPlotHistograms.java plot



8:32:03 AM ---------- compile successful

classpath:/org/freehep/jas/extension/compiler/web/examples/CreateAndPlotHistograms.java 8.75/12.7MB

BROOKHAVEN NATIONAL LABORATORY

JAS Examples - Fitting

JAS3

File Edit View Tuple Run Window Help

Programs

Examples × Fit.java ×

```
1  import hep.aida.*;
2  import java.util.Random;
3
4  public class Fit
5  {
6    public static void main(String[] args)
7    {
8      // Create factories
9      IAnalysisFactory  analysisFactory = IAnalysisFactory.create();
10     ITreeFactory     treeFactory = analysisFactory.createTreeFactory();
11     ITree            tree = treeFactory.create();
12     IPlotter         plotter = analysisFactory.createPlotterFactory().create("Fit.java Plot");
13     IHistogramFactory histogramFactory = analysisFactory.createHistogramFactory(tree);
14     IFunctionFactory functionFactory = analysisFactory.createFunctionFactory(tree);
15     IFitFactory      fitFactory = analysisFactory.createFitFactory();
16
17     IHistogram1D h1 = histogramFactory.createHistogram1D("Histogram 1D",50,-3,3);
18
19     Random r = new Random();
20
21     for (int i=0; i<100000; i++) {
22       h1.fill(r.nextGaussian());
23       h1.fill(r.nextDouble()*10-5);
24     }
25
26     IFunction gauss = functionFactory.createFunctionFromScript("gauss",1,"background+a*exp(-(x[0]-mean)*(x[0]-mean)/sigma/sigma)","a,m
27
28     gauss.setParameter("a",h1.maxBinHeight());
29     gauss.setParameter("mean",h1.mean());
30     gauss.setParameter("sigma",h1.rms());
31
32     plotter.region(0).plot(h1);
33     plotter.region(0).plot(gauss);
```

```
8:34:11 AM ---------- compile successful
```

classpath:/org/freehep/jas/extension/compiler/web/examples/Fit.java 8.22/20.3MB

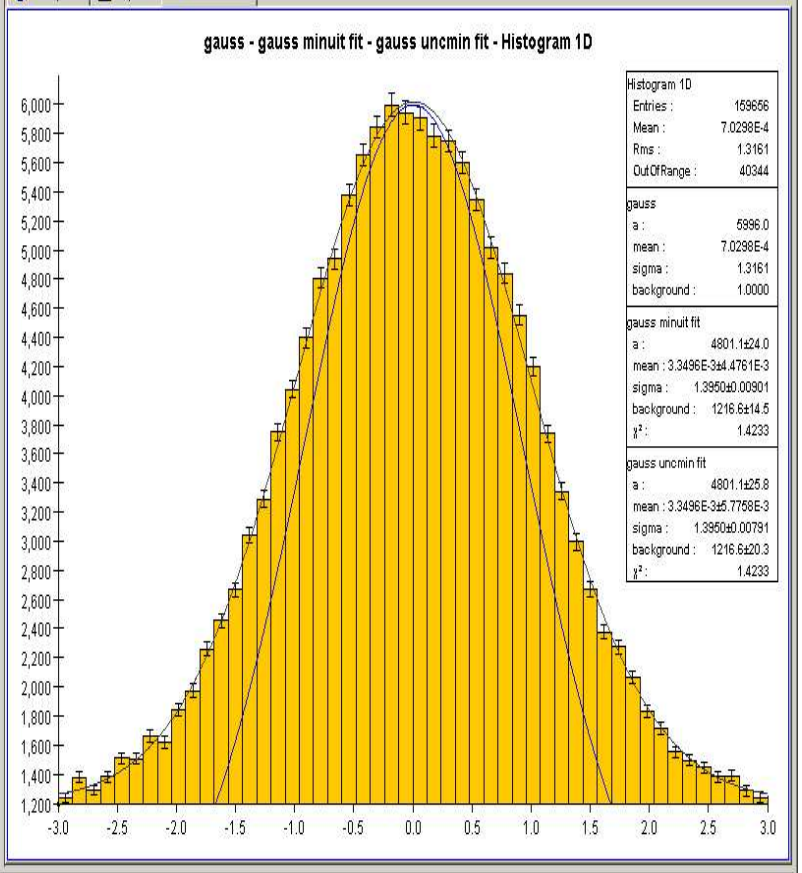
JAS3

File Edit View Tuple Run Window Help

Programs

tree-0

Examples × Fit.java × Fit.java Plot ×



Compiler × Fit ×

classpath:/org/freehep/jas/extension/compiler/web/examples/Fit.java 13.7/20.3MB

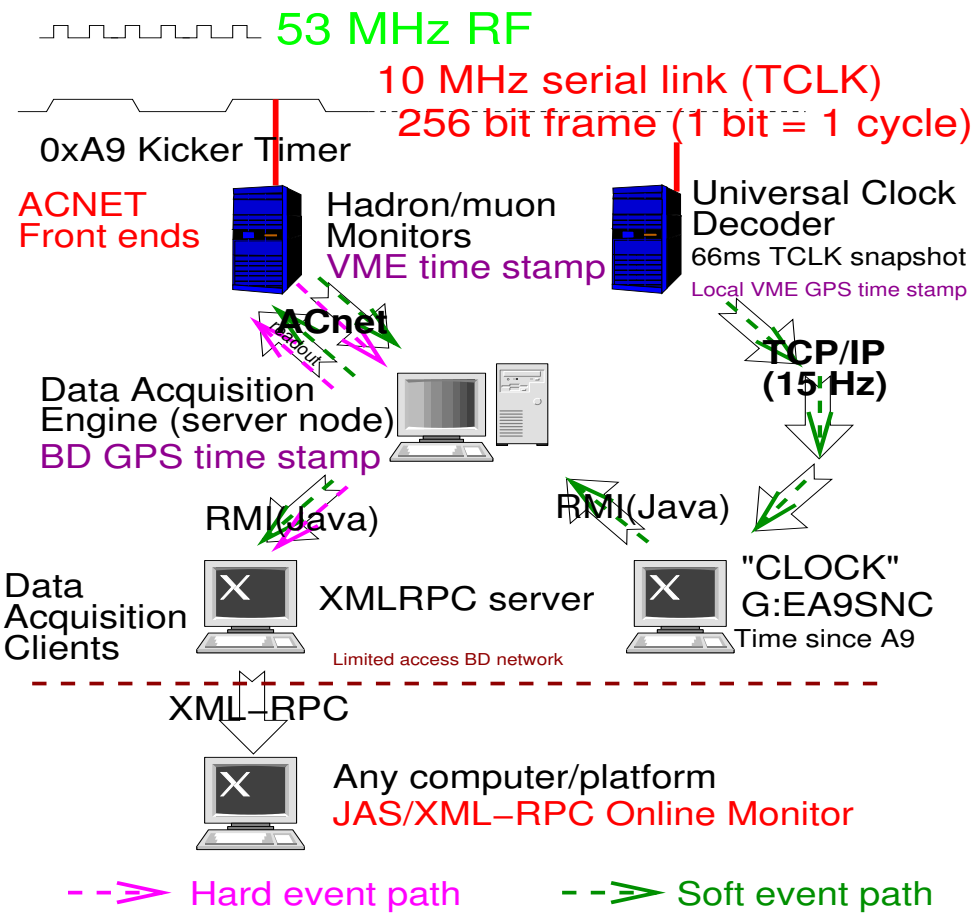
BROOKHAVEN NATIONAL LABORATORY

Why Use JAS for NuMI Monitoring?

- Java GUI capabilities well established and easy to use.
- Java XML-RPC implementation = access to ACNET data.
- JAIDA allows handling of complex data, full HEP data analysis capabilities including histogramming, fitting and ntupling.
- JAS provides nice GUI interface to JAIDA for run-time histogram and fitting displays.
- JAS3 architecture is based on FreeHEP Application Framework into which many optional modules can be plugged in. Highly customizable.
- JAS/JAVA already in use by FNAL AD controls (Timofei Bolshakov, Jim Patrick and others) so local expertise available.
- Real reason: Too lazy, already knew JAVA, and hate ROOT/C++(M. Bishai).

BROOKHAVEN NATIONAL LABORATORY

Getting NuMI Data → JAS Monitor



- A request is sent via HTTP to Charlie Kings' XML-RPC service to extract the data from a list of ACNET devices 1 second after an A9 accelerator event (NuMI kicker fires) occurs.
- The "clock" process listens to a 15 Hz broadcast of TCLK snapshots. "clock" informs the DAE when the snapshot contains the A9 event.
- The DAE requests the data from the ACNET front ends and sends it to the XML-RPC server which broadcasts it to the world

Any computer running Java can monitor ACNET data using JAS/XML-RPC: http://minos.phy.bnl.gov/~bishai/minos/NuMIMon/

BROOKHAVEN NATIONAL LABORATORY

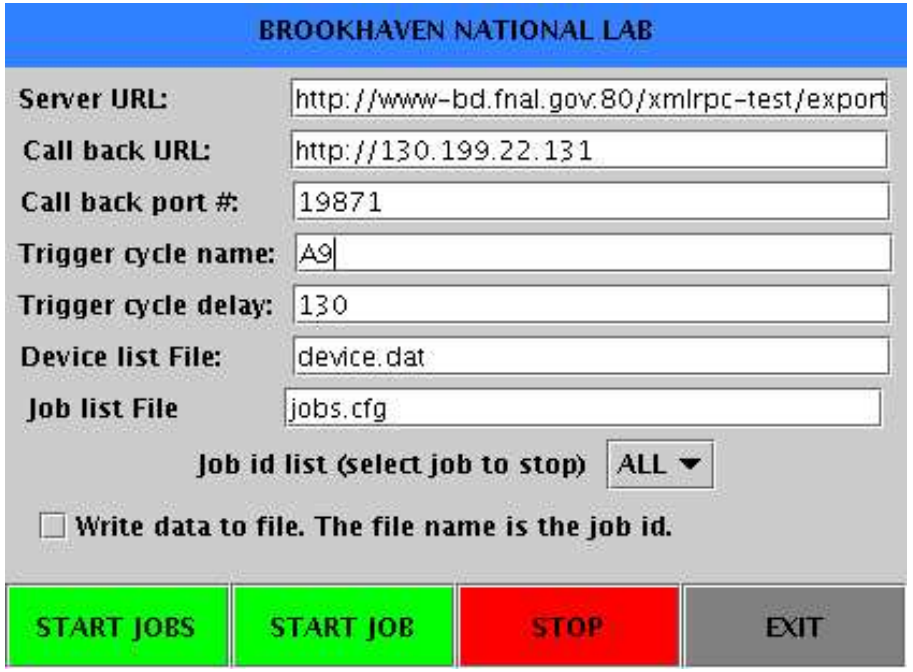
LIVE DEMO OF NuMI JAS/XMLRPC MONITOR (time and network permitting)

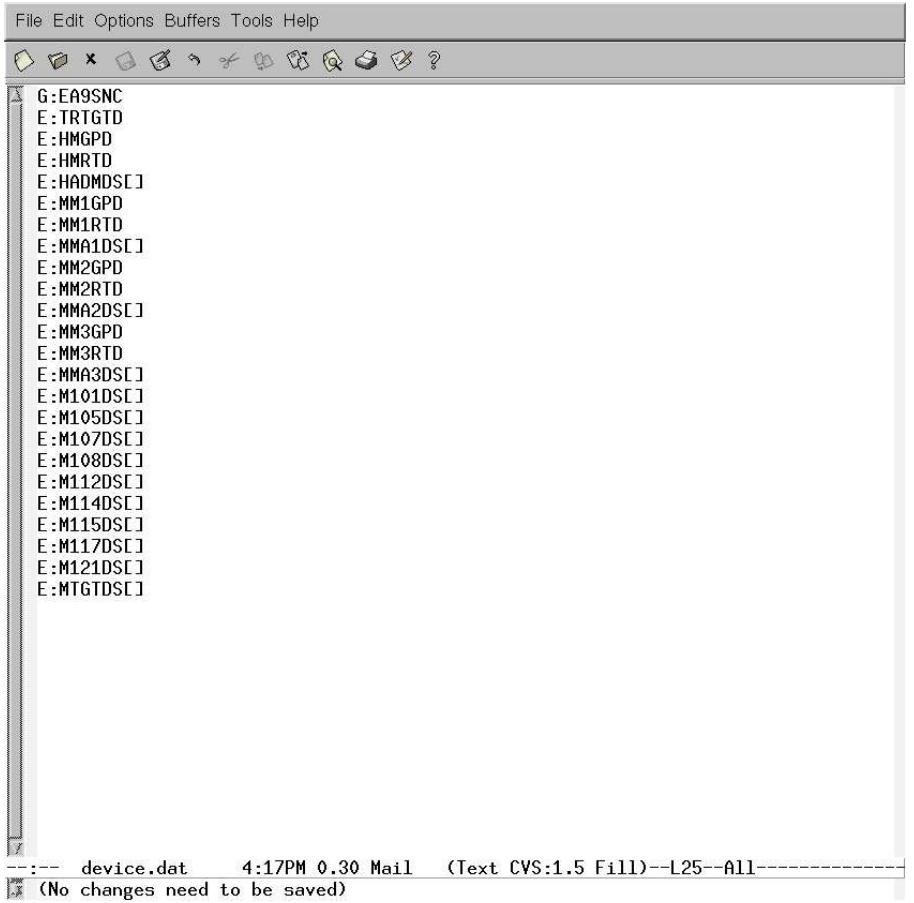
If not - show screen snapshots

BROOKHAVEN NATIONAL LABORATORY

NuMIMon Screen Shots

The XML-RPC interface dialog box:

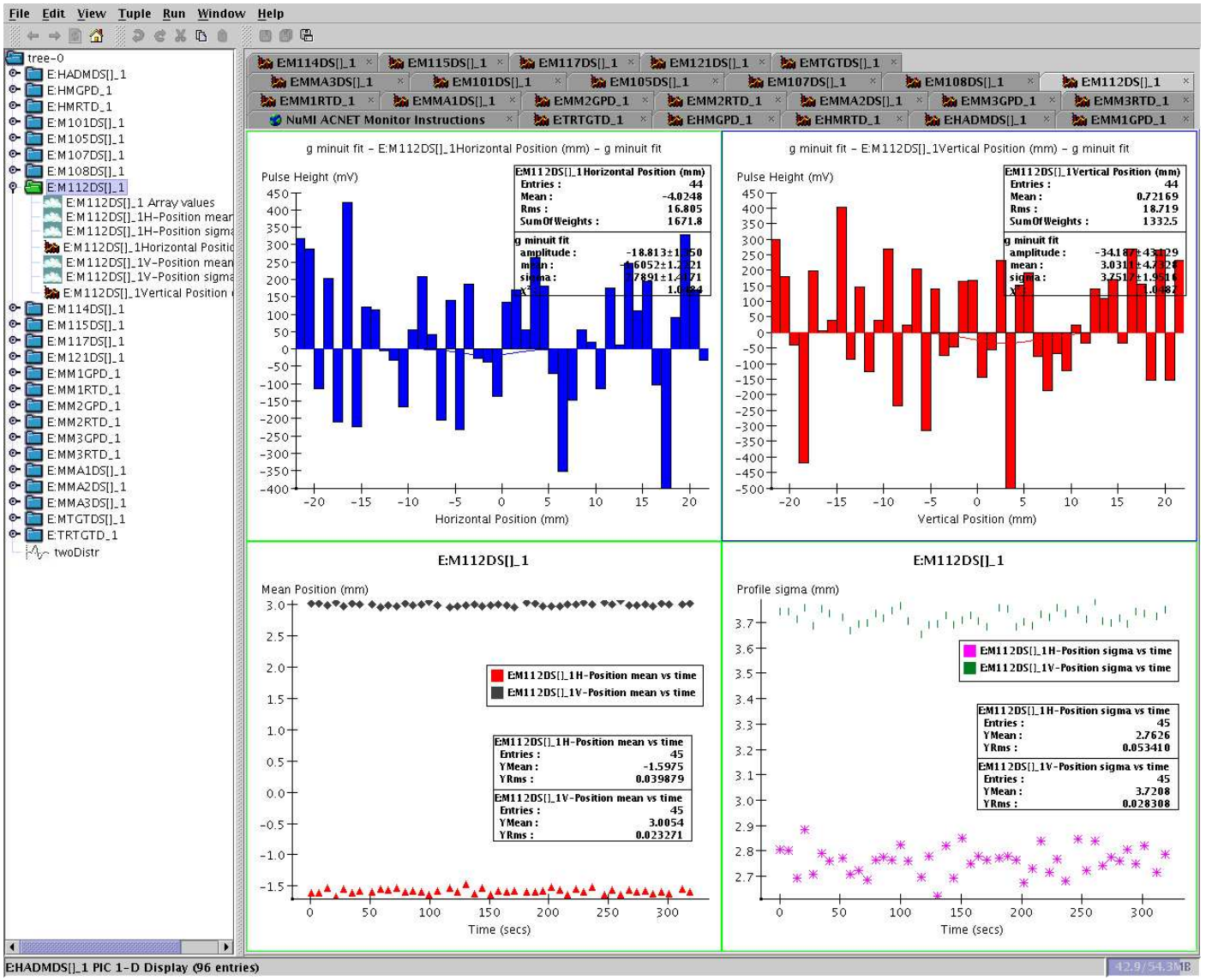




NuMIMon Screen Shots

BROOKHAVEN NATIONAL LABORATORY

NuMIMon Screen Shots



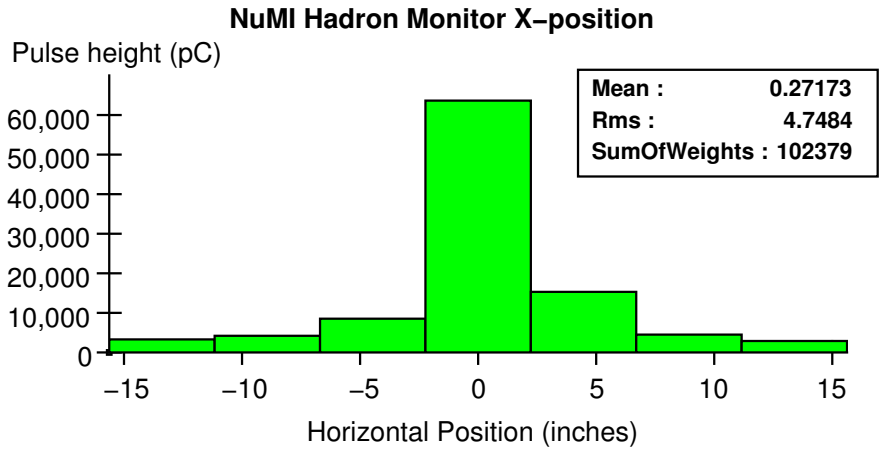
BROOKHAVEN NATIONAL LABORATORY

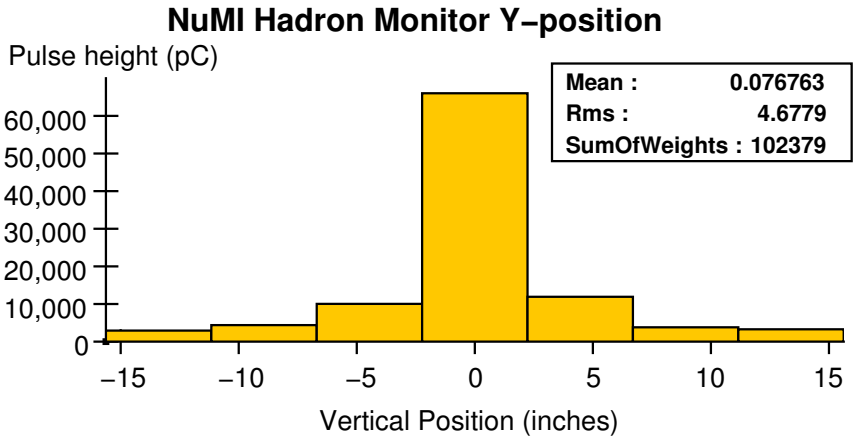
NuMIMon Data Dump

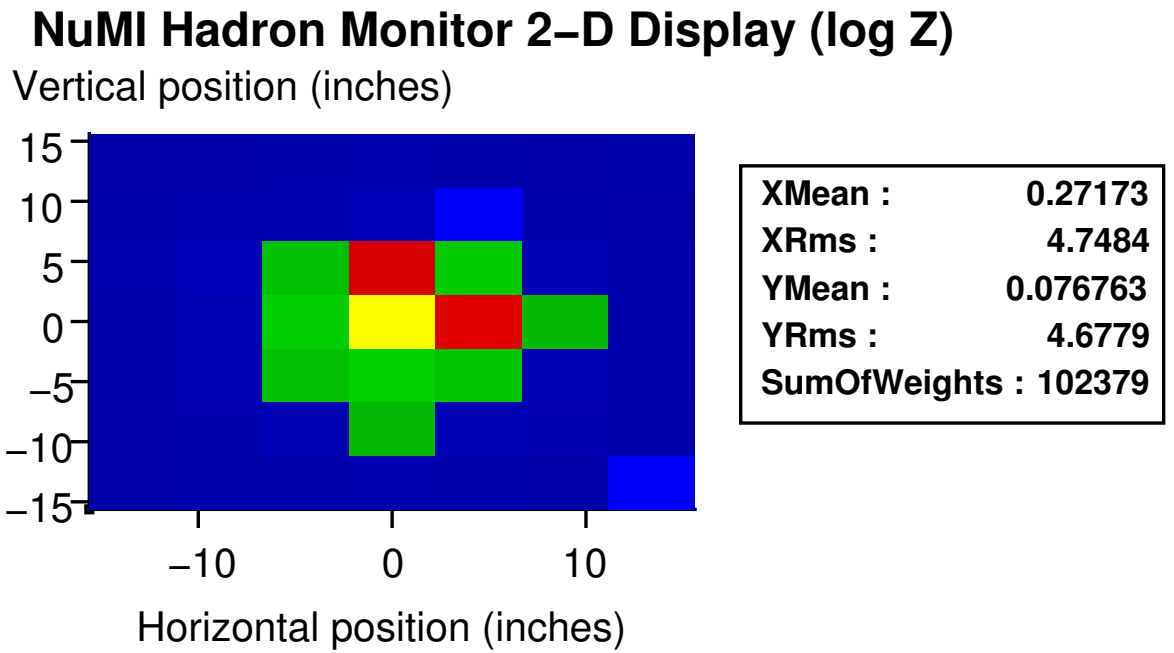
```
mary /cygdrive/c/Program Files/JAS3/bin
Start-time Sat Dec 04 21:45:11 EST 2004
TCLK-cycle A9
Cycle-delay 1000
Device-count 15
E:HMGPD 1102214720960 0 778.75 TORR
E:HMRTD 1102214720960 0 59.46875 DegF
E:HADMDS[] 1102214720962 1102214719917 -1.83  -0.305  -1.22  -0.915  0.915  -0.305
1.525  -0.0  -1.22  0.305  0.915  -1.83  -0.0  1.22  -0.915  1.22  -0.915  0.305  -0
.61  -1.525  -0.0  0.915  -0.305  0.61  -1.525  -1.22  -0.305  -0.61  -0.61  0.305
-2.44  0.915  -0.61  0.305  -0.61  -0.915  0.305  0.915  -1.22  -2.44  -0.61  -0.915
  1.83  0.61  0.915  1.22  -1.525  -0.61  -0.0  -2.44  -0.305  0.61  0.305  0.61  -0
.305  -3.05  0.915  -2.745  -1.22  -1.525  0.305  -1.22  -0.61  -1.22  -2.135  -0.91
5  -0.0  -0.61  -1.22  0.915  -1.83  -2.135  0.305  0.61  -1.22  -0.61  -0.915  -0.6
1  -0.305  -0.61  0.305  0.305  -1.525  -0.915  -1.22  -1.22  0.915  -0.915  1.22  -
2.135  1.22  -0.61  -1.22  -0.305  0.915  -2.44
E:MM1GPD 1102214720960 0 778.1875 TORR
E:MM1RTD 1102214720960 0 58.62500000000001 DegF
E:MMA1DS[] 1102214720962 1102214719924 -48.19  -110.715  -12.504999999999999  -65.27
  -63.44  -12.504999999999999  -71.065  98.21  -71.37  -48.495  -47.58  10.674999999
999999  -44.225  14.334999999999999  -61.61  7.93  -25.314999999999998  -0.0  -41.17
5  32.025  -35.074999999999996  21.96  -50.019999999999996  -7.015  103.395  -49.714
999999999996  22.265  -17.69  -51.85  27.45  -46.055  -100.345  85.705  17.384999999
999998  -1.525  42.089999999999996  -14.03  -13.42  -33.55  3.965  -7.32  1.22  11.5
9  16.775  9.76  27.145  -53.375  -7.32  -40.26  96.99  -63.44  21.349999999999998
-38.125  13.42  -20.74  55.815  -67.405  -10.674999999999999  154.32999999999998  7.
625  -132.67499999999998  -119.255  83.875  122.0  224.17499999999998  114.07  -62.8
3  28.06  -40.565  21.96  18.91  58.56  39.65  -35.685  -38.125  -130.23499999999999
:
```

BROOKHAVEN NATIONAL LABORATORY

NuMI First Beam !





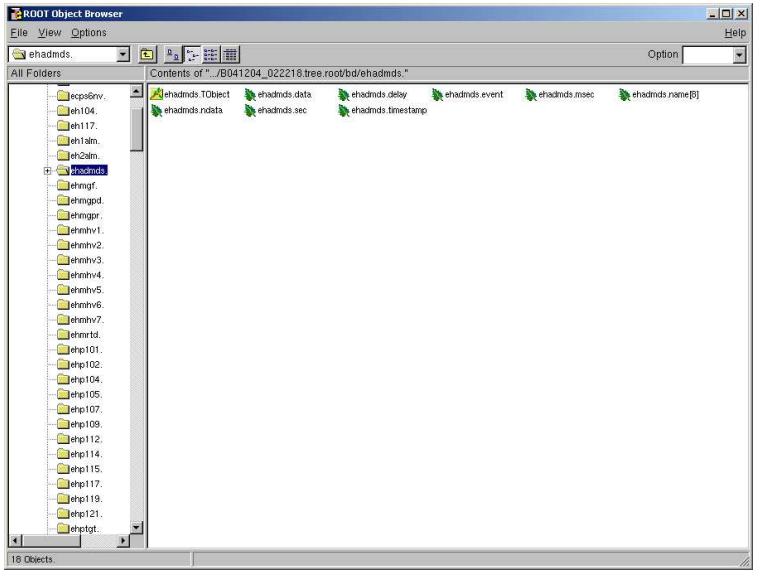


BR
NATIONAL LABORATORY

OFFLINE ANALYSIS OF NUMI DATA FROM DEC 3-4th.

Accessing NuMI Data Offline

- Charlie King's XML-RPC server → BeamData process → minos-acnet.fnal.gov → archive (Bretts talk).
- From archive, Brett converts raw data to a Root TTree.
- Used standalone Root macros to analyze TTree.
- Analyzed BPM, profile monitor and muon/had monitors.



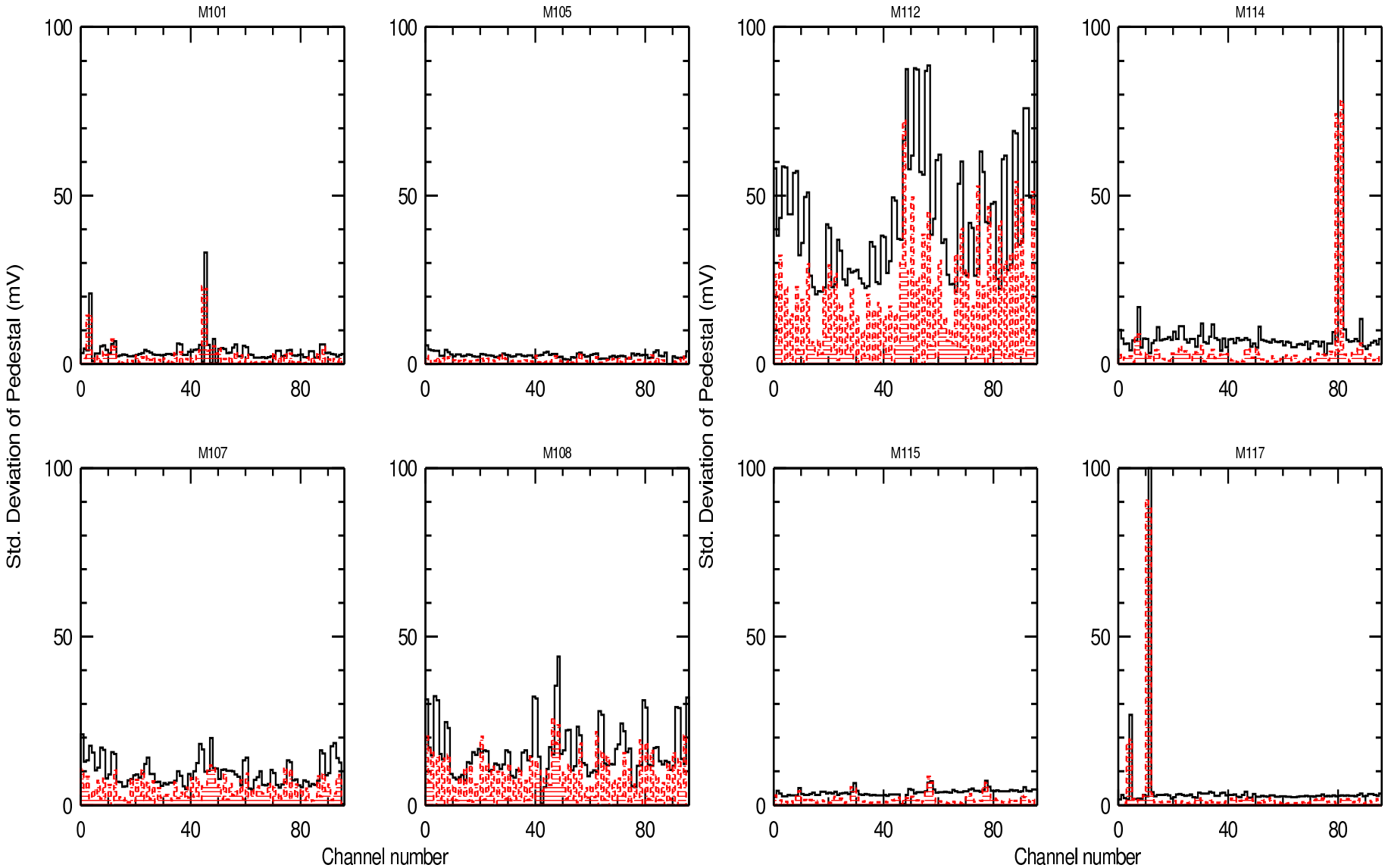
BROOKHAVEN NATIONAL LABORATORY

Profile monitor analysis

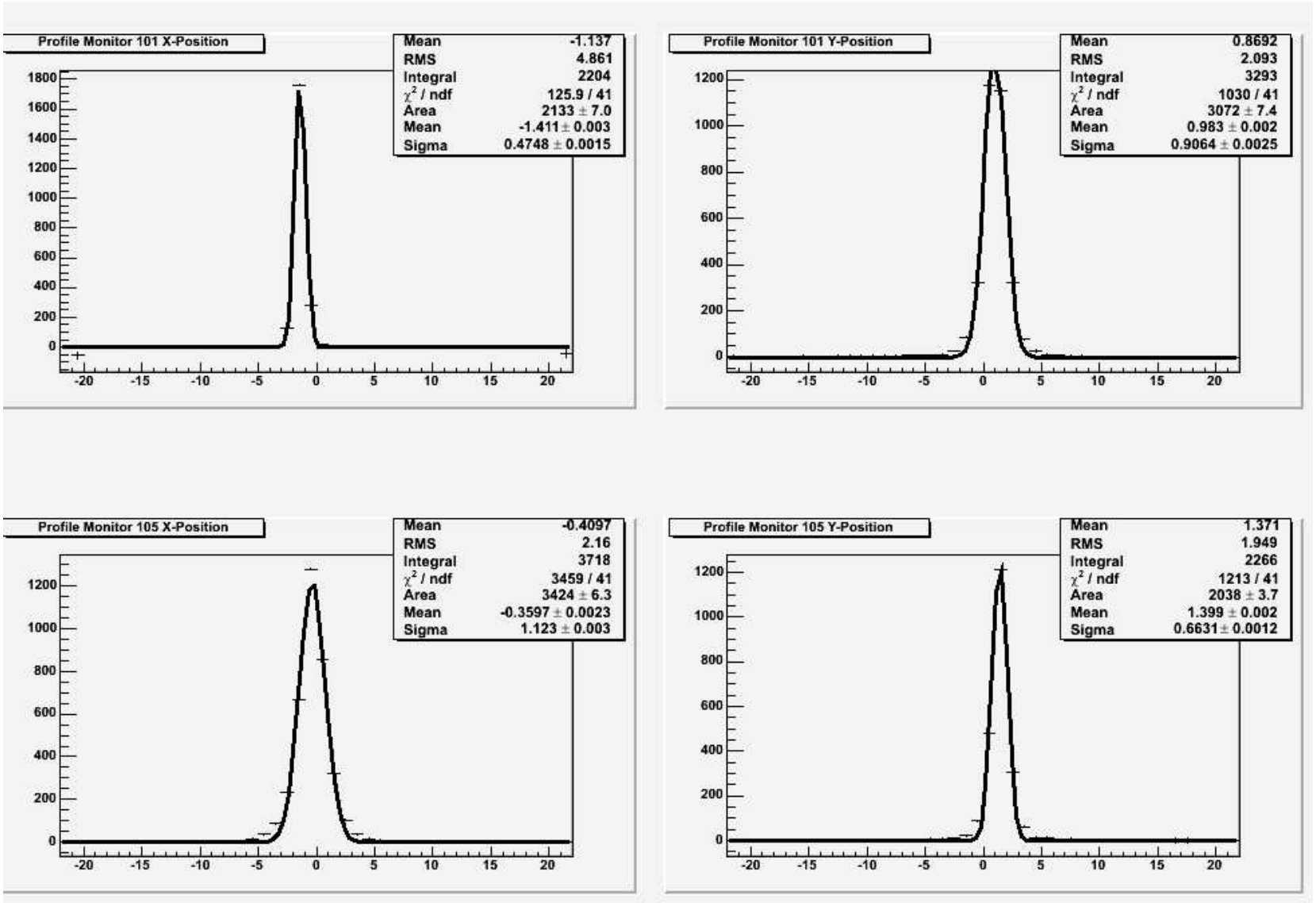
- Step 1: Calculate pedestals from events with v. low beam intensity as measured in TORTGT and/or TOR101.
- Step 2: Calculate noise = $\sigma(P_n)$, where P is the pulseheight in channel n.
- Step 3: Calculate differential noise = $\sigma(P_n - P_{n+1})$ to identify common mode noise and cross-talk.
- Step 4: Loop over events with beam intensity > 1×10^{11} and fit profiles in each spill to a single Gaussian using $\sigma(P_n)$ as the uncertainty on the measurement in each channel.
- Step 5: Compare profile widths with beam emmitance expectations (P. Lucas).
- Step 6: Compare means with nearby BPMs (Sacha, Mark D.)

BROOKHAVEN NATIONAL LABORATORY

Profile monitor noise/dnoise

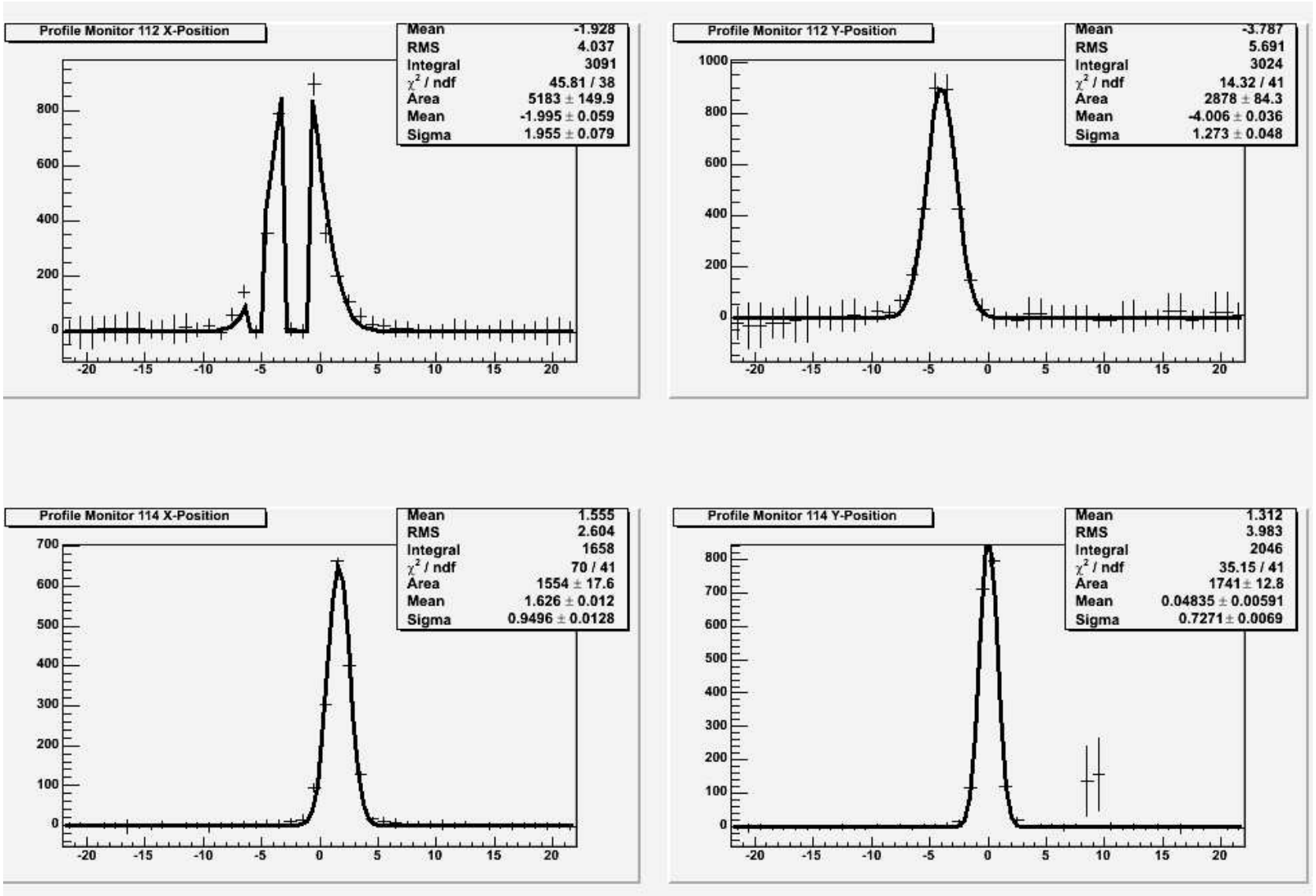


Profile/Had Monitors - Spill 43



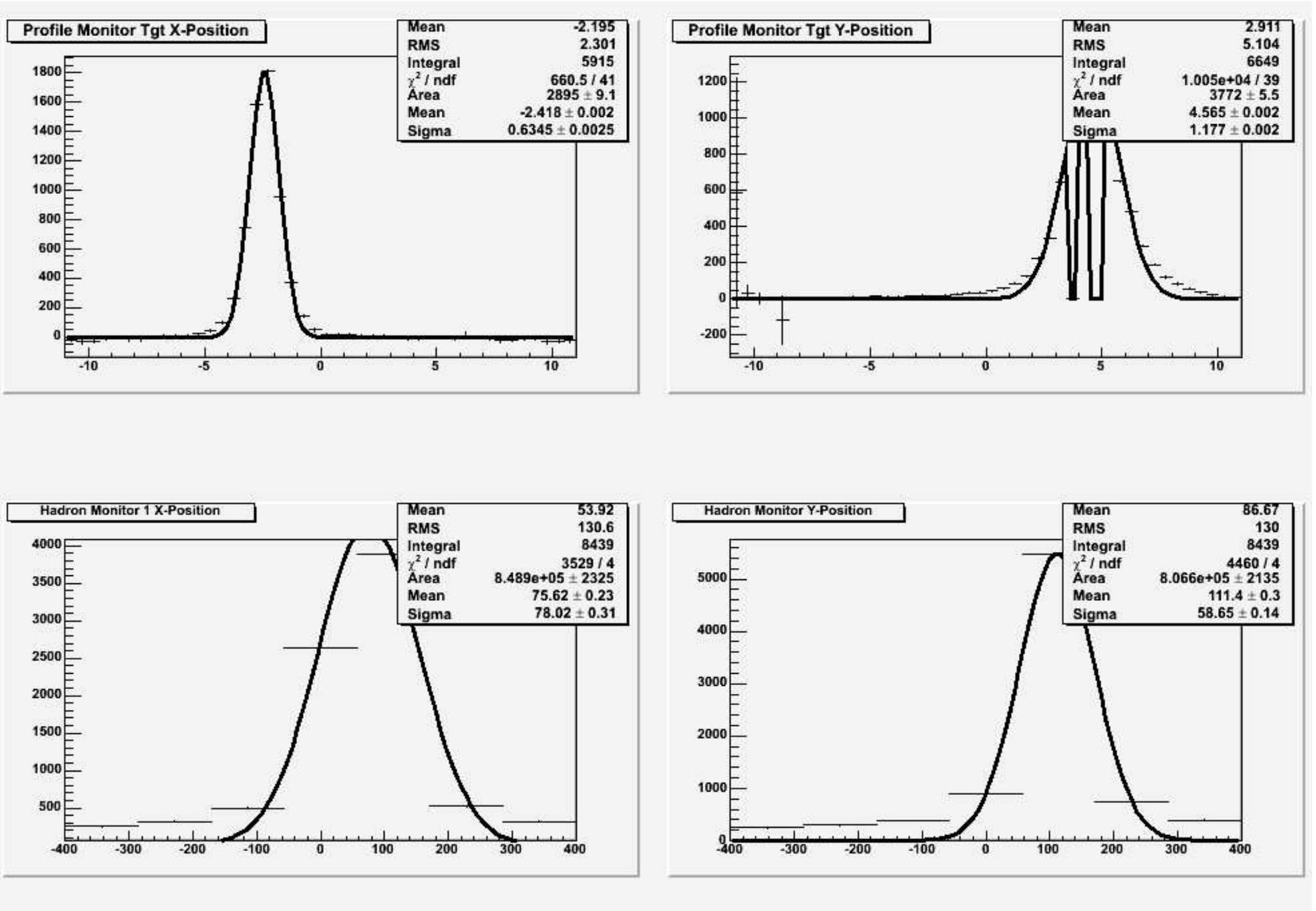
BROOKHAVEN NATIONAL LABORATORY

Profile/Had Monitors - Spill 43



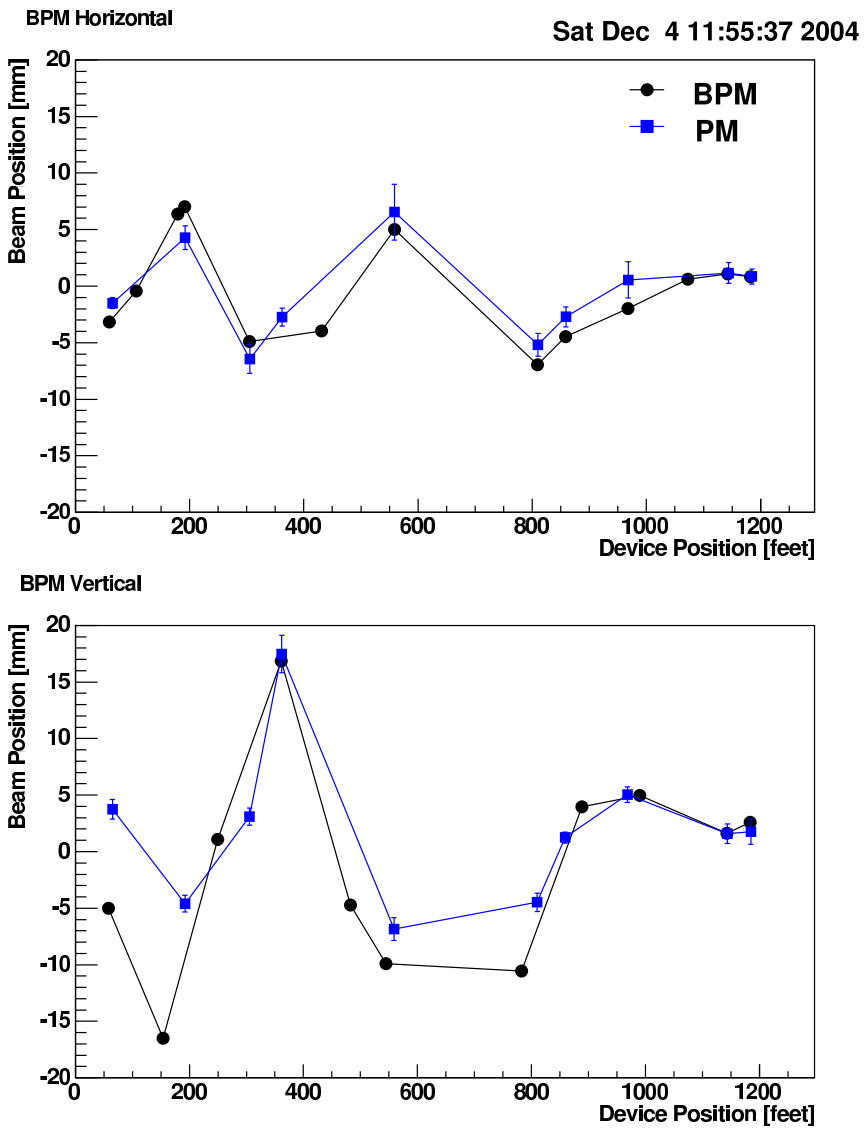
BROOKHAVEN NATIONAL LABORATORY

Profile/Had Monitors - Spill 43

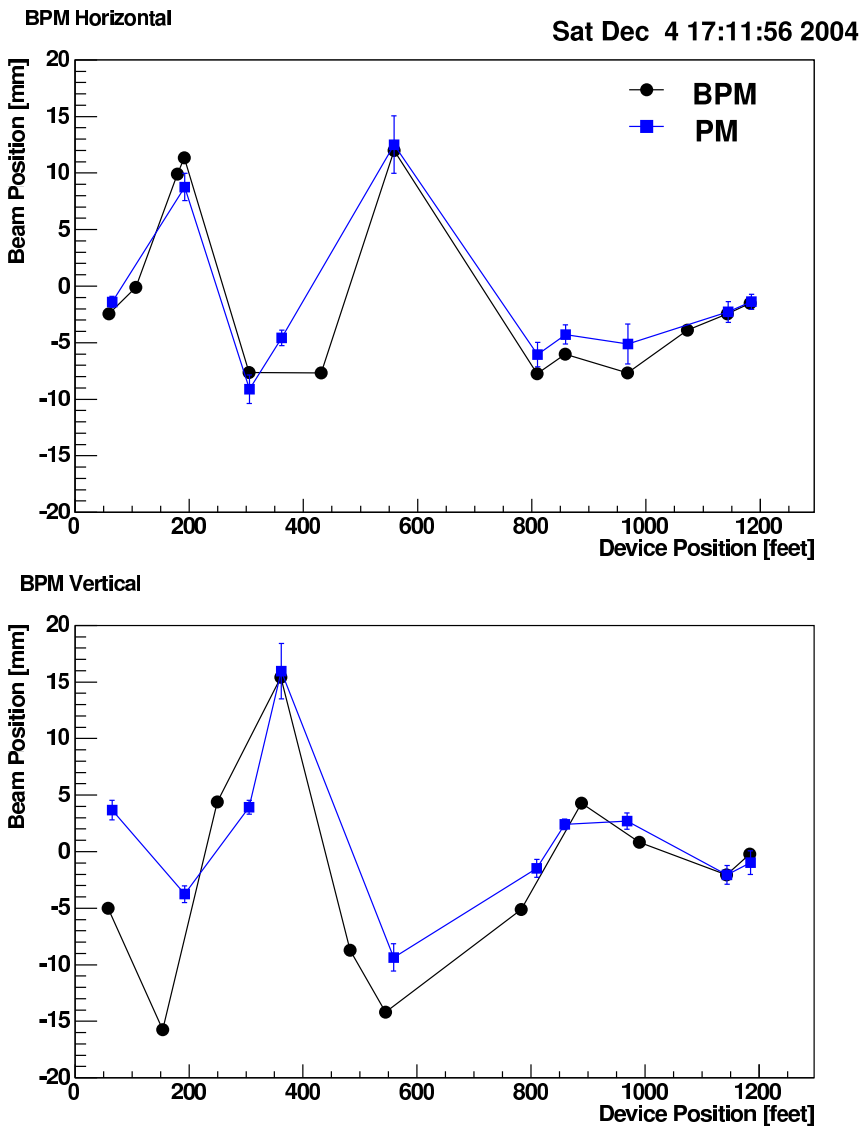


BROOKHAVEN NATIONAL LABORATORY

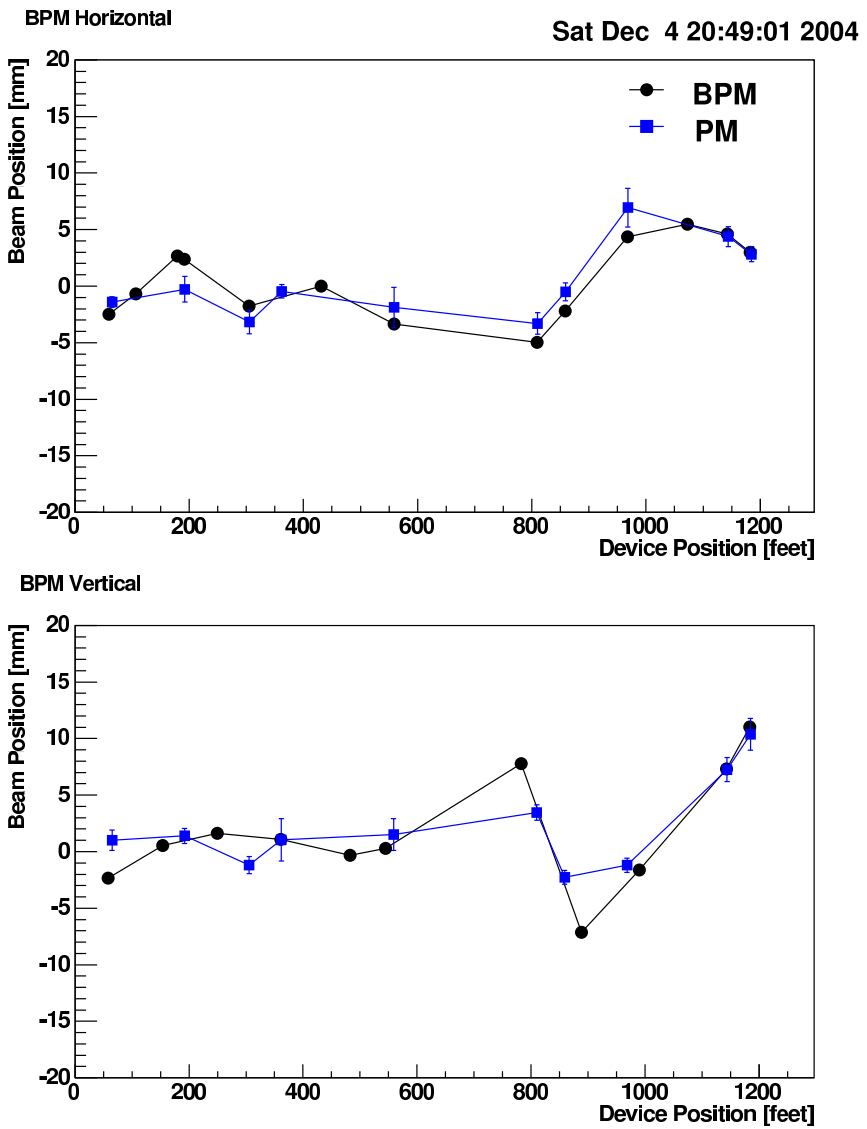
Profile monitor vs BPMs



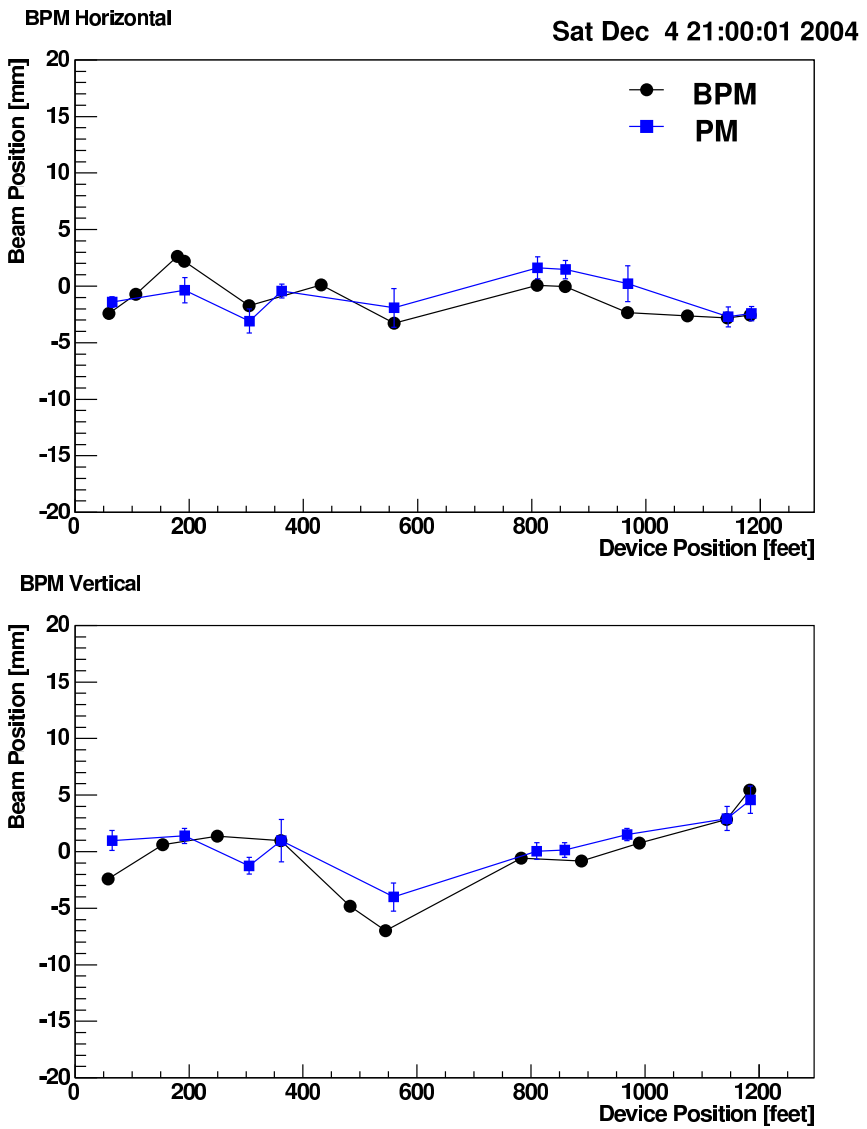
Profile monitor vs BPMs



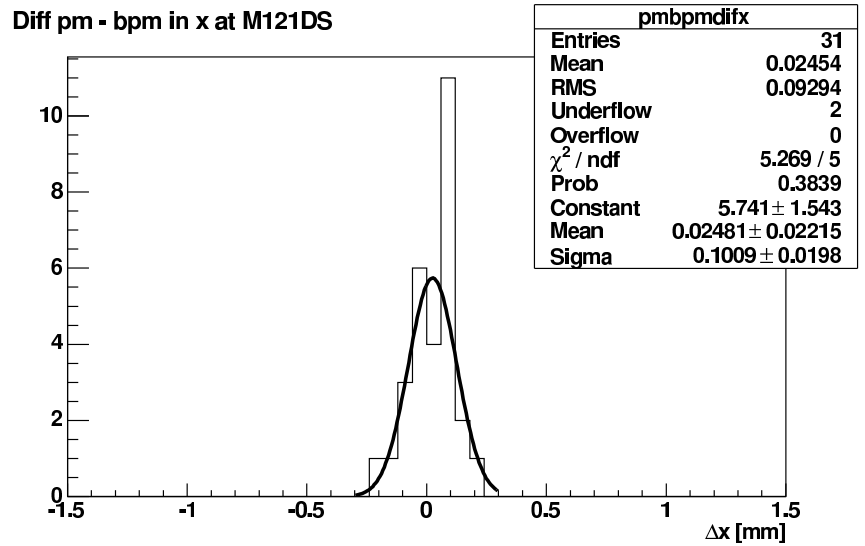
Profile monitor vs BPMs

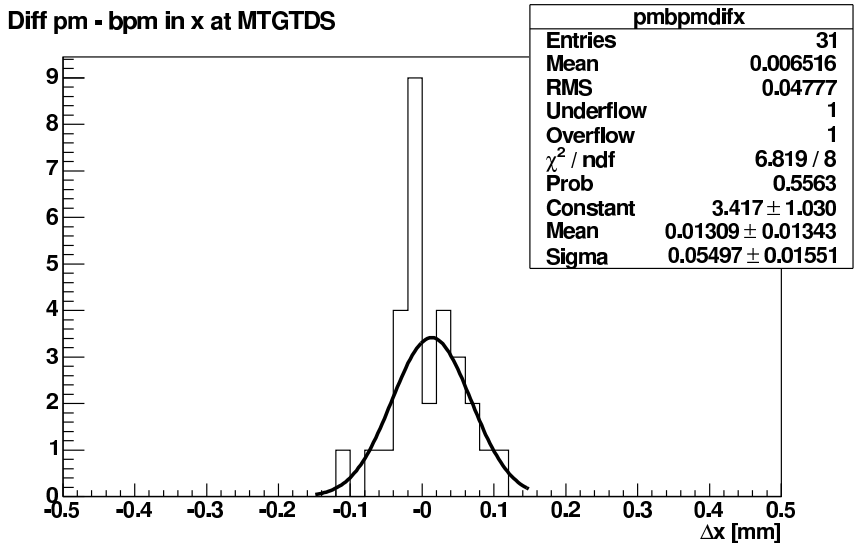


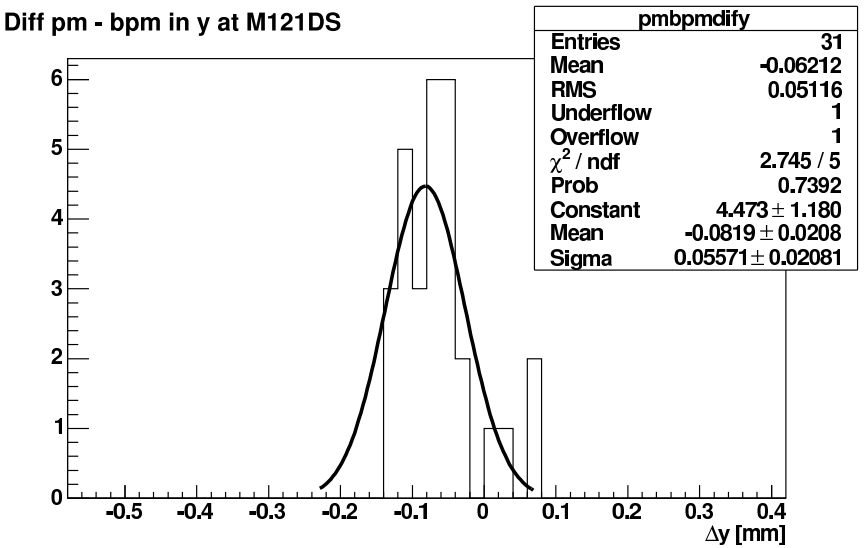
Profile monitor vs BPMs

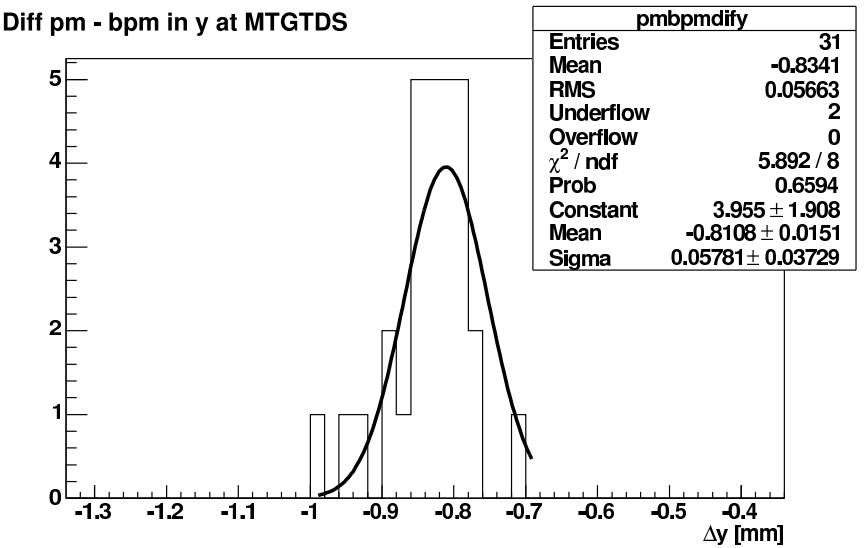


Profile monitor vs BPMs





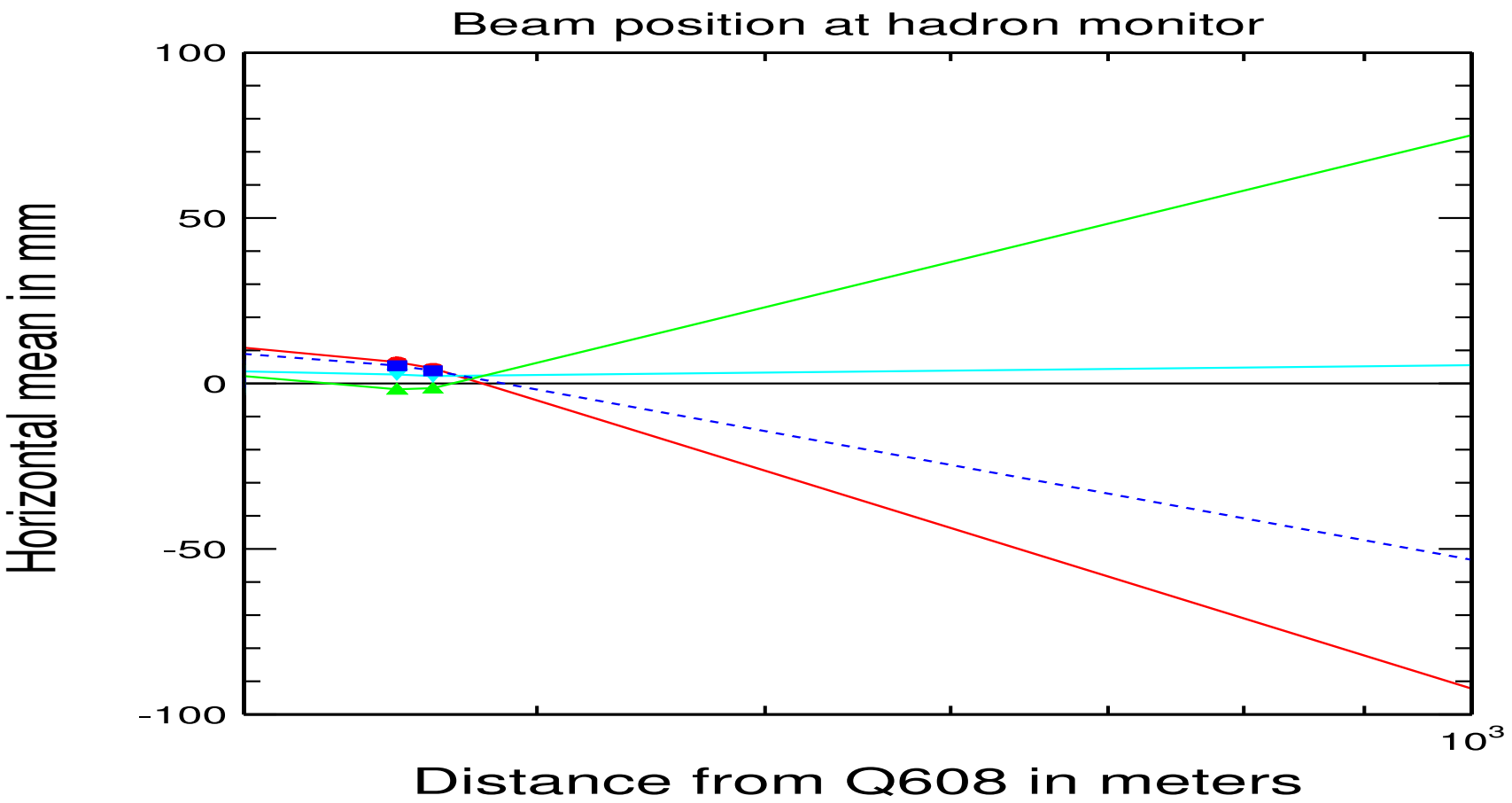




BROOKHAVEN NATIONAL LABORATORY

Hadron Monitor tracks PM

See Sacha's results for a full blown quantitative analysis



Problems: Timing Pathologies

An analysis of the timestamps from the 14 SWICs revealed the following ACNET data timing pathologies in the 385 spills taken:

1. The DAE timestamp of PM114,115,117,121,TGT was 20 seconds ahead of the rest of the SWICS. VME timestamps were identical to ± 30 msecs: 1 spill.
2. The VME timestamps are not in sync on all 14 SWICS. The event contained data from the previous spill on some devices: 10 consecutive spills. Problem limited to PM114,115,117,121,TGT.
3. No data was returned from PM101,105,107,102,112 and we got an error from the DAE or all zero data: 3 spills.

NB: Charlie K. has a new service which may have already fixed these problems.

BROOKHAVEN NATIONAL LABORATORY

Conclusions

- The JAS3 online NuMI monitor worked well in real conditions. Flexible enough to add new devices and new analysis code and plots on the fly. Some problems with Java memory management/stability. Need to implement a shifter oriented error warning system.
- The ACNET XML-RPC service worked well during data taking. Some timing problems with accessing the data. Its not clear (to me) that the problem is with the server and not further upstream.
- The BeamData process was very easy to run, very reliable and analyzing the NuMI offline data was a blast → if you know some ROOT!.

BROOKHAVEN NATIONAL LABORATORY