Beam Quality Parameters and Protons-on-Target using the IF Beam DB

Lisa Goodenough, ANL

Joint Beam Group/NuMI-X Working Group Session
NOνA Collaboration Meeting
Fermilab
April 4, 2014

An accurate determination of the number of protons-on-target (POT) requires an understanding of the condition of the NuMI beam on a spill-by-spill basis.

The new Intensity Frontier Beam Database (IF Beam DB) stores the readout values from hundreds of devices monitoring various components of the beam line.

We use several of these devices to assess beam quality for a given spill and thus determine whether the POT for that spill should be included in the total POT count and also whether the spill data should be analyzed.

IF Beam DB

- Homepage for IF Beam Data Server is
  http://dbweb0.fnal.gov:8080/ifbeam/app/event_monitor

- From users' perspective, IF Beam DB consists of data retrieval "bundles" that one can access in various ways.

  ➙ A direct web query can be made. For more information, see https://cdcvs.fnal.gov/redmine/projects/ifbeamdata/wiki/DataAccessSyntax.

  ➙ Database can be queried using the Offline framework with C++ libraries included in NOνASoft.

- Contents of bundles are user-defined, *i.e.* users choose the device information stored in the bundles. NOνA utilizes the NuMI_Physics_A9 bundle, currently a collection of ~60 devices deemed to be useful for doing physics.

IFDBSpillInfo_module

- NOνASoft module responsible for collecting data related to the beam on a spill-by-spill basis
- Accesses device information from the IF Beam DB
- Calculates the physical parameters of the beam used for making beam quality cuts
- Compares beam parameters with cuts to determine whether the spill had "good" beam
- Stores the results in the "reco" ROOT files

Beam Quality Parameters and Cuts

- POT (min POT cut: 0.50×10^{12})

  The POT for a spill is the first non-negative toroid value from the following devices in this order: TRTGTD, TR101D. If both values are negative, indicating that the beam was off, then a value of "0" is assumed.

- horn current (-205 kA < horn current < -195 kA)

  The horn current is calculated using the 4 horn stripline peak current values appropriately shifted and normalized (*i.e.* calibrated).

- horizontal and vertical beam position at the target (0.02 mm < pos x(y) < 2.00 mm)

  The beam position at the location of the target is calculated using beam position and beam position intensity monitors. Values from devices positioned at two different locations in the pre-target region, so-called "121" and "TGT", are used to extrapolate the position to the target itself.

- horizontal and vertical beam width at the target (0.57 mm < width x(y) < 1.58 mm)

  The horizontal and vertical spread of the beam at the target location are calculated using the beam profile monitors.

- time difference between event time and database time ($\Delta t < 1 \times 10^{9}$ nsec)

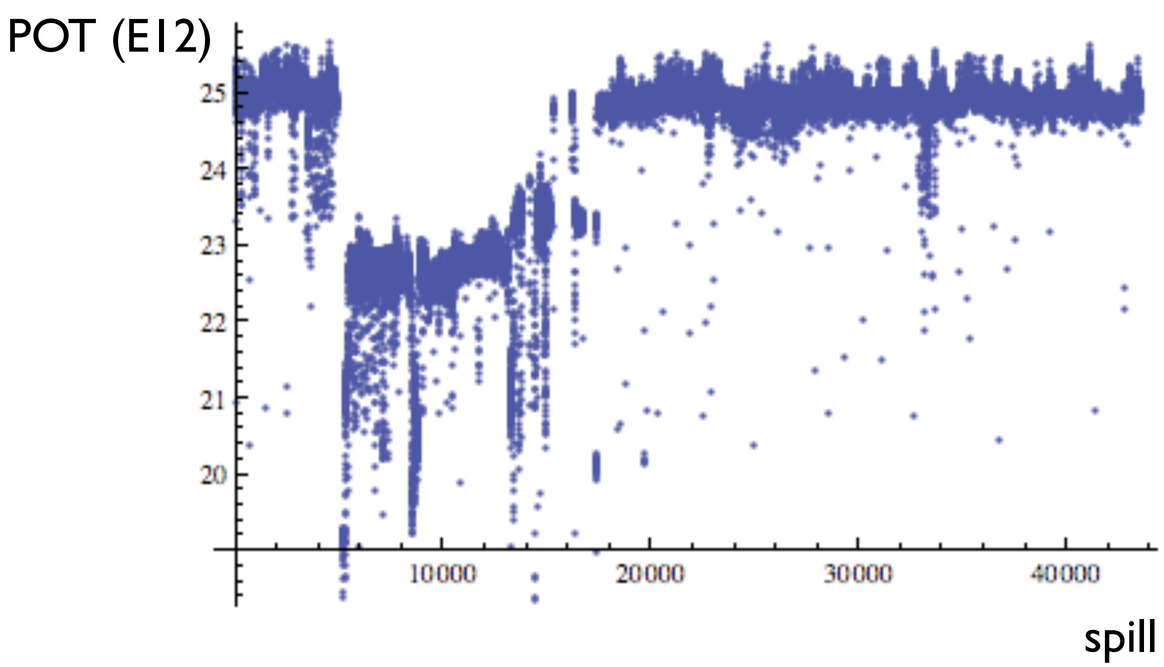
  The difference between the event time and the database time is used to verify that the device values obtained were the correct values for that event.

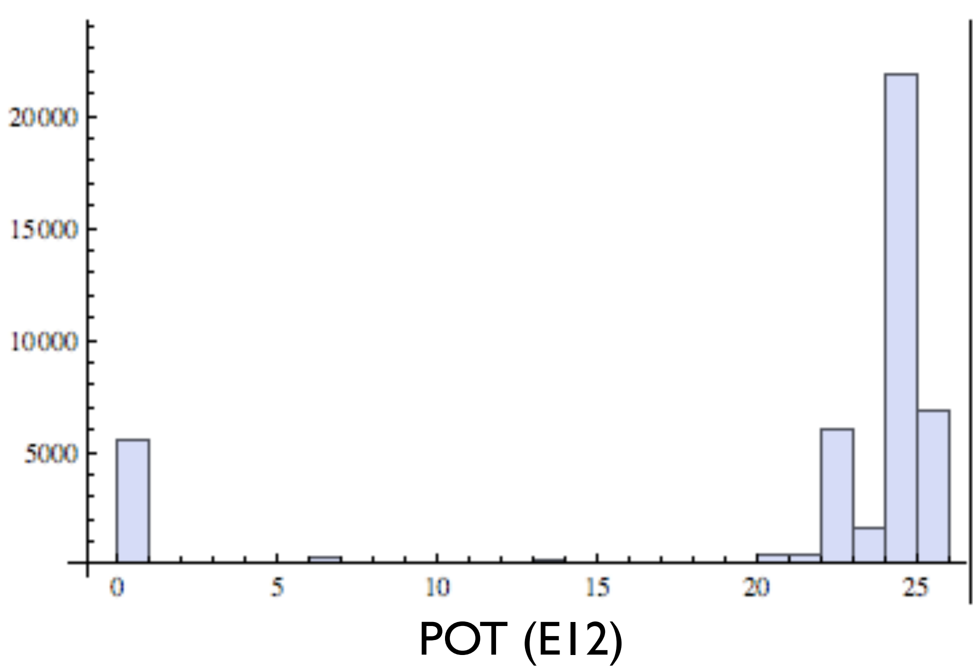
POT

- POT for a spill is first non-negative toroid value from the following devices in this order: TRTGTD, TR101D. If both values are negative, indicating "Beam Off", then a value of "0" is assumed.
- I have analyzed several days' worth of POT data spanning several months and have concluded that the noise ceiling is 0.02E12 for TRTGTD and 0.01E12 for TR101D.
- We take a conservative value for the POT cut for determining beam quality and use 0.5E12 for both devices. Any spill with POT less than 0.5E12 is not considered "good".

POT

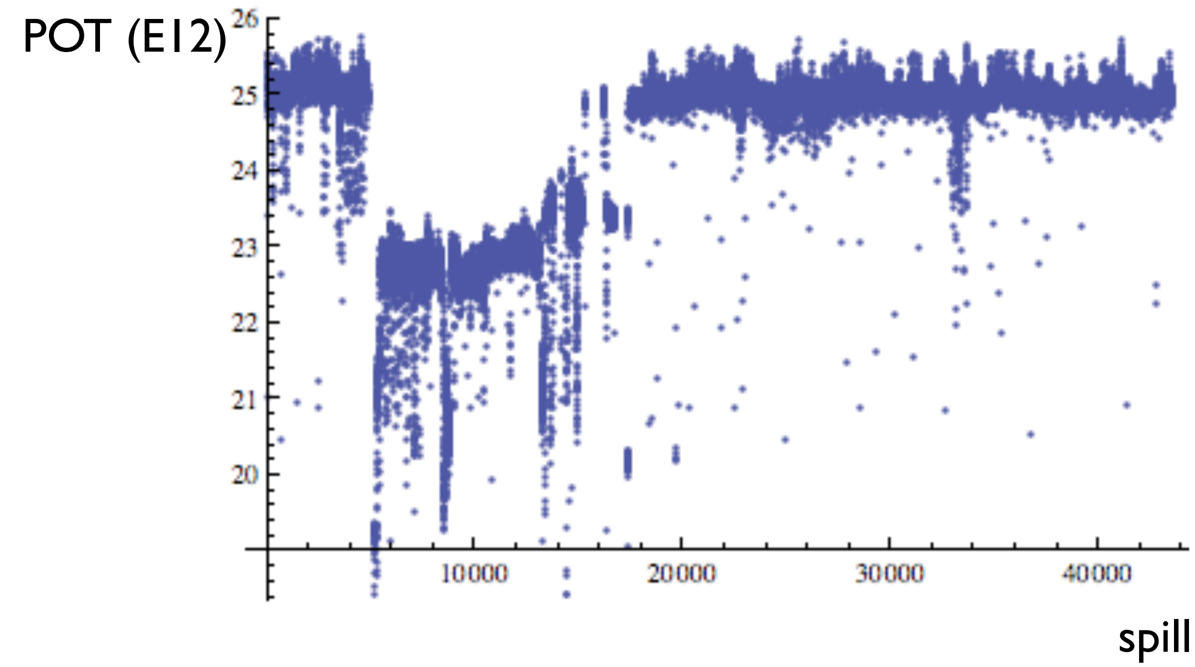
TRTGTD
values over 24 hour period

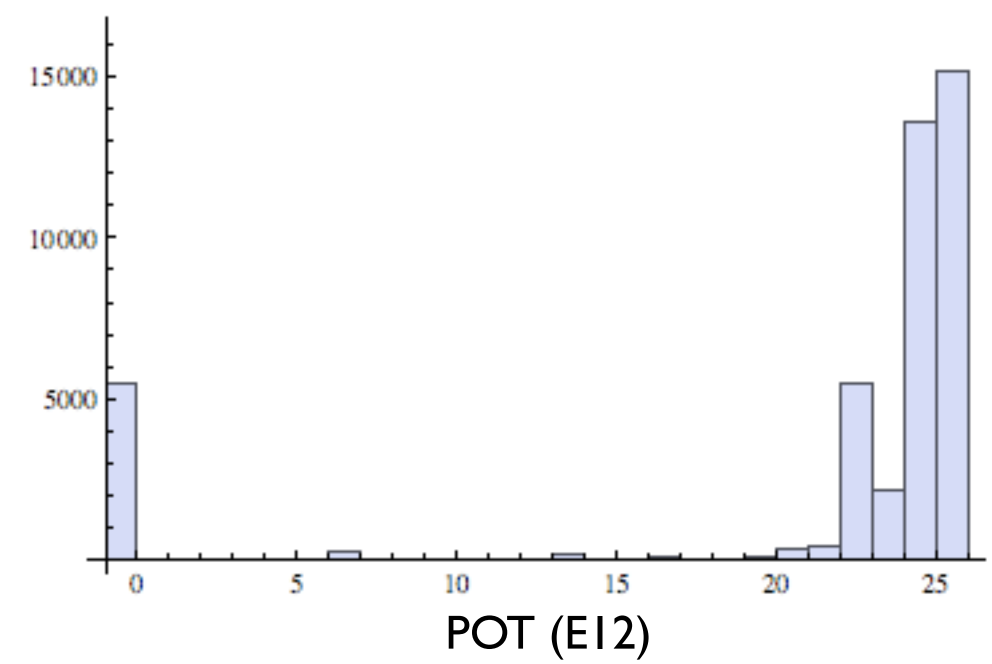




no values between 0.02E12 and 0.82E12

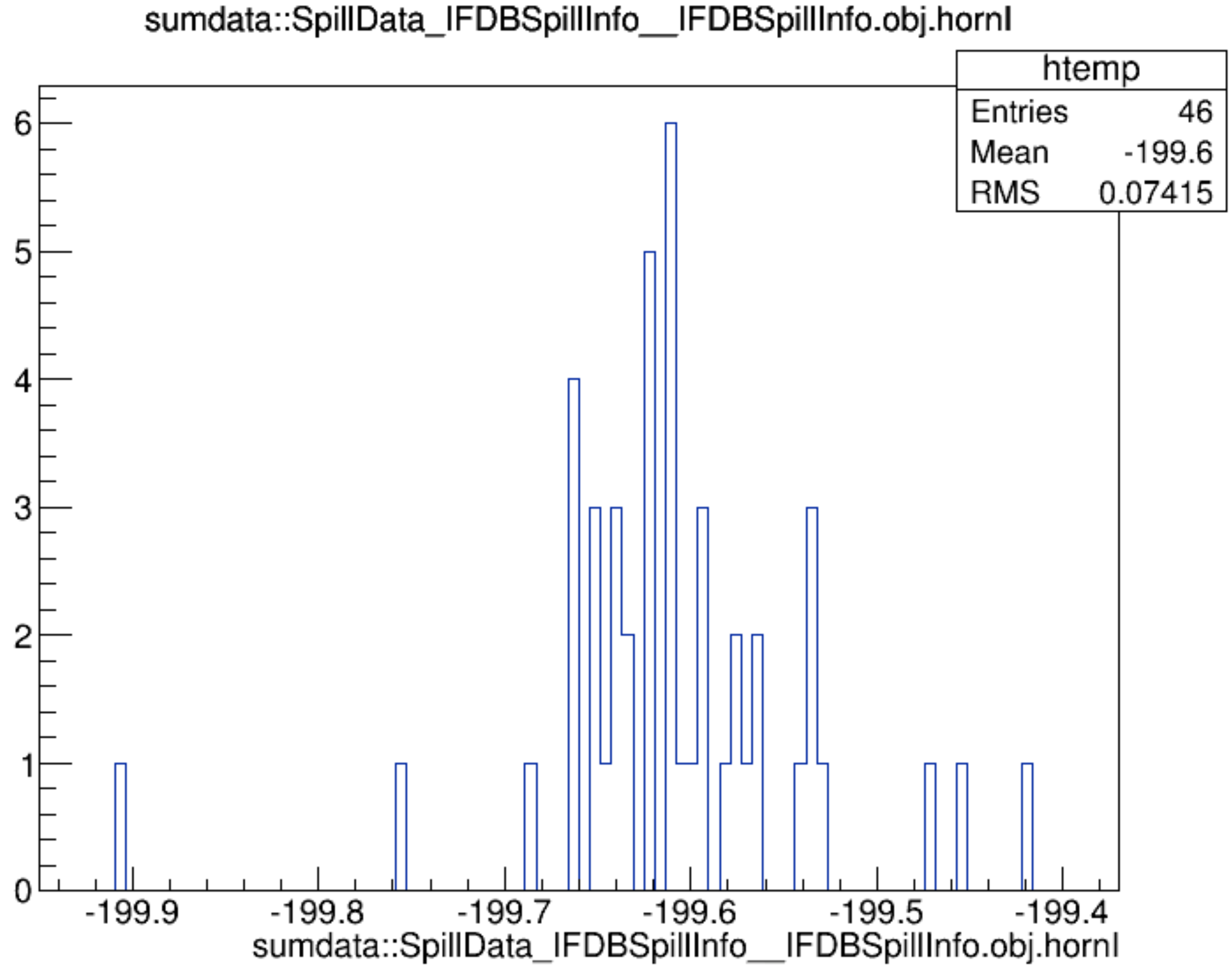
TR101D
values over 24 hour period





no values between 0.0E12 and 0.80E12

Horn Current

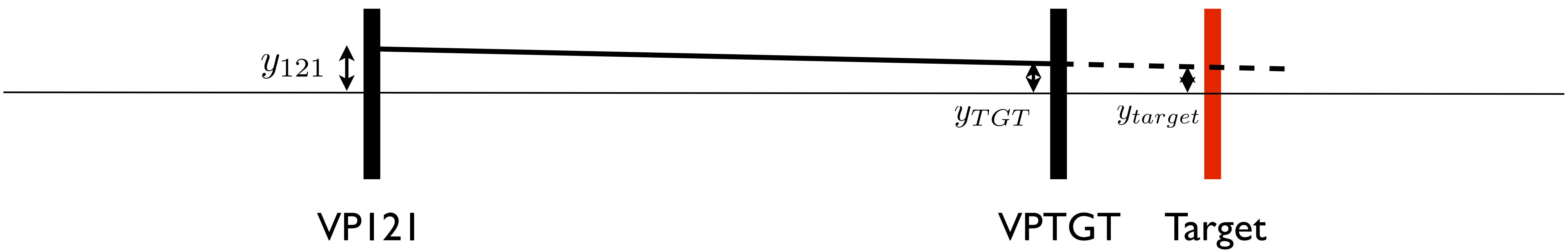


The horn current should be within the range:
-205 kA < horn current < -195 kA

Beam Position at the Target

The horizontal and vertical beam positions at the target are calculated using beam position and beam position intensity monitors.

Values from devices HP121, VP121 and HPTGT, VPTGT positioned at pre-target locations "121" and "TGT" are used to make a linear extrapolation to the position of the target itself.

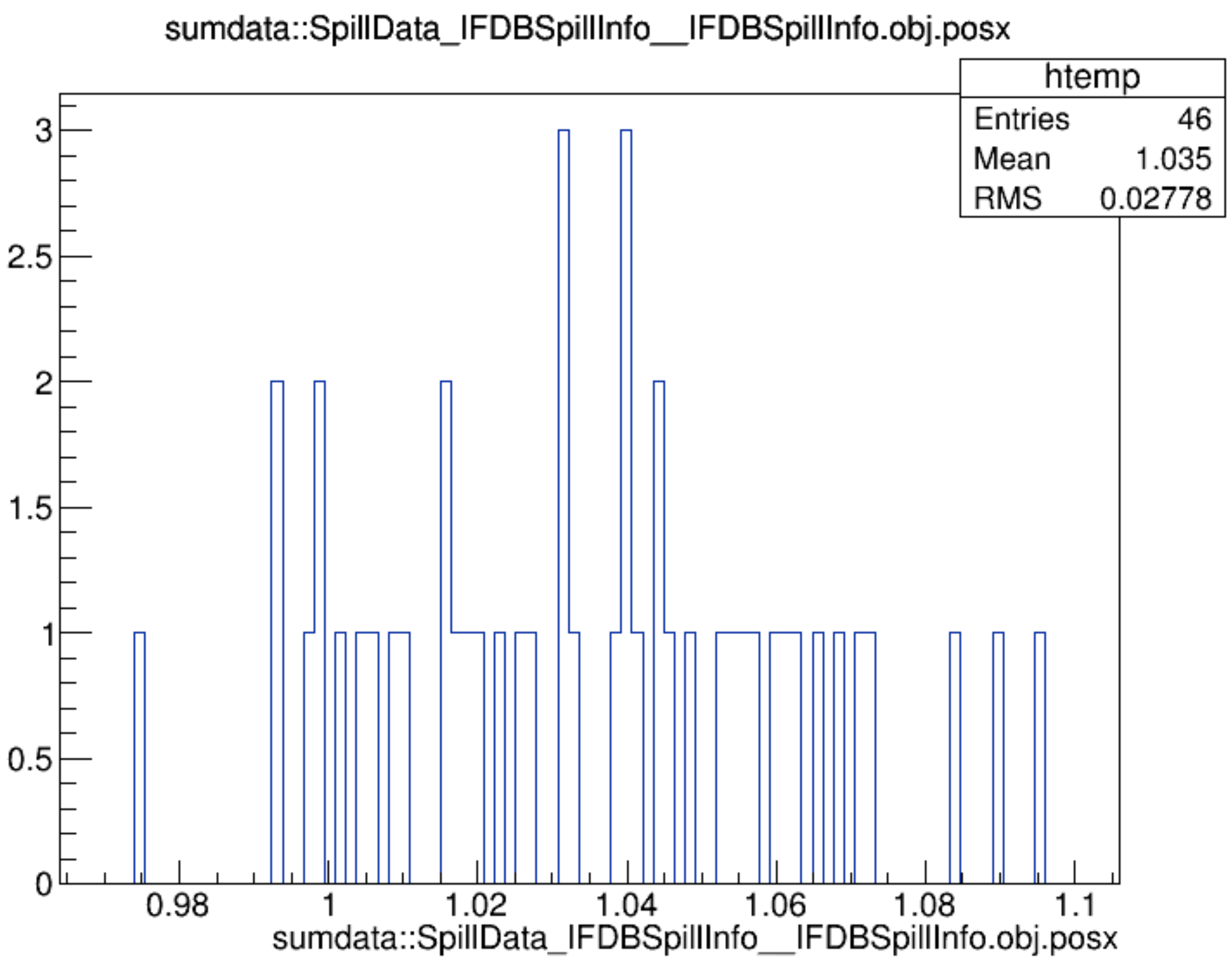


Beam position and beam position intensity monitors measure values for each of the six batches in the Main Injector. We calculate the beam position for each batch and take the average value.

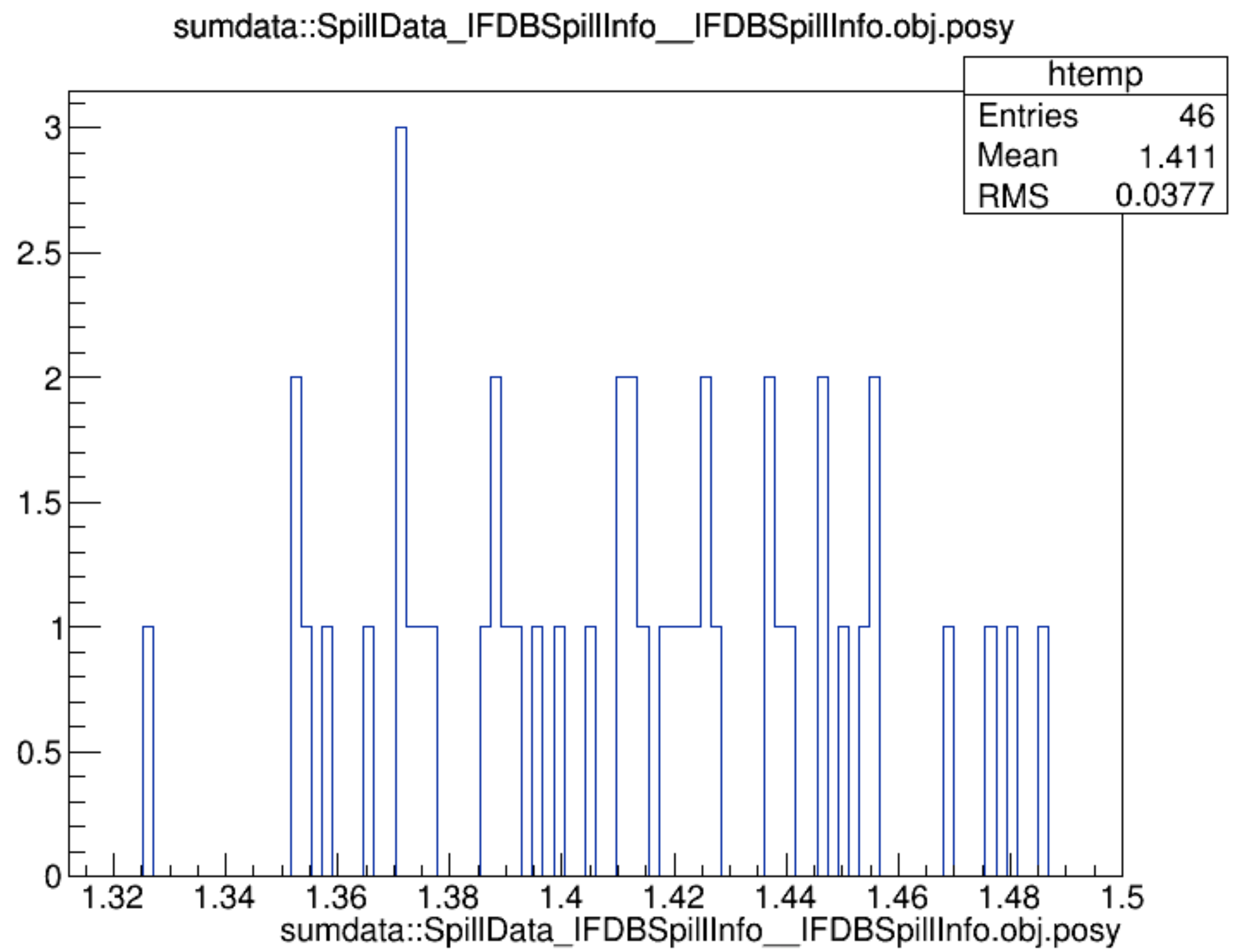
Note: the device values for HITGT and VITGT (the intensity values for each batch) for times prior to December 3rd, 2013 are currently not stored in the DB, but will (hopefully) be added to the DB soon. Once these values are present for all times, we will use the intensity-weighted average position value using HITGT and VITGT as the weighting values.

Beam Position at the Target

horizontal position



vertical position



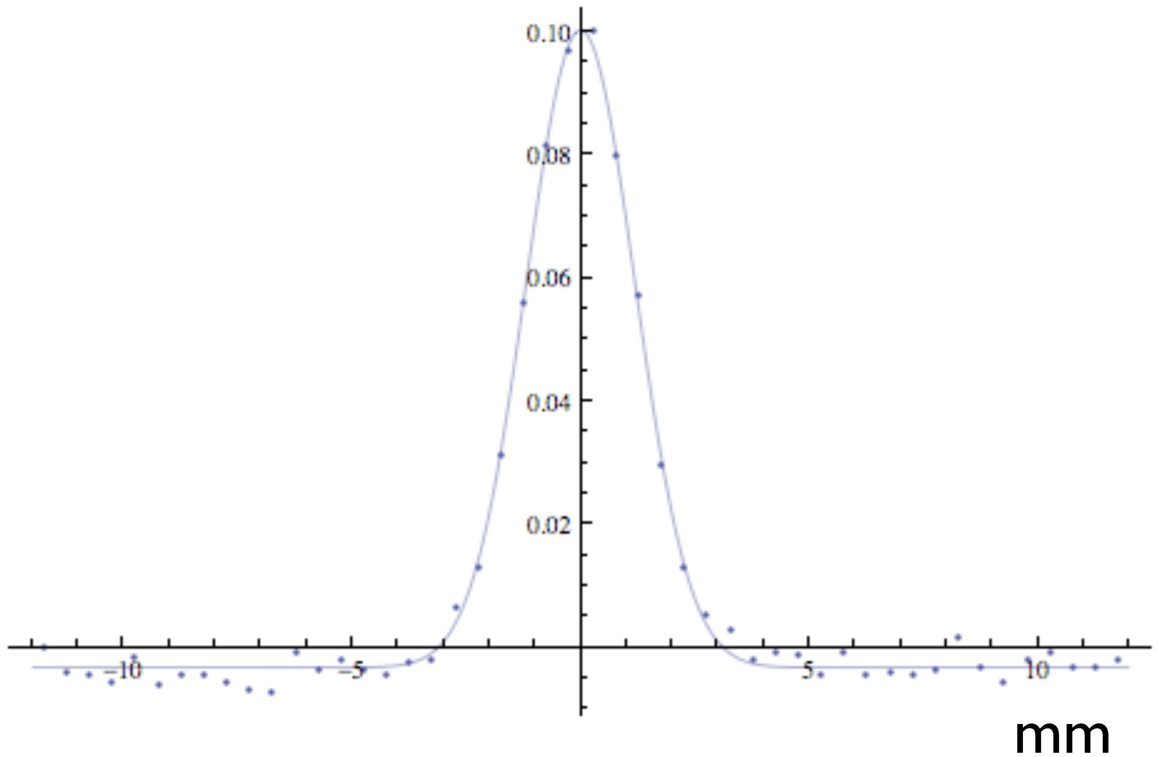
The horizontal and vertical positions of the beam should be within the range:
0.02 mm < posx(y) < 2.00 mm

Beam Width at the Target

- Beam width at the target is calculated using data from the beam profile monitors.
- Horizontal and vertical profiles are separately fit to a Gaussian distribution + offset. Beam widths are defined as widths of the Gaussians.
- In the MINOS era, widths at the pre-target locations "121" and "TGT" were used to extrapolate the width to the target location.
- However, beam monitor "121" is not in the beam line continuously anymore. It is inserted once daily at 2:30PM for 15 minutes, and according to Phil Adamson is "more or less useless". There is a new beam profile monitor ready for installation at "121", which, after installation, will be in the beam continuously.
- Present method of calculating the widths is different - beam width at target is defined as the width at the location of "TGT". Once the new BPM has been installed, we will revert to extrapolating using data from "121" and "TGT".

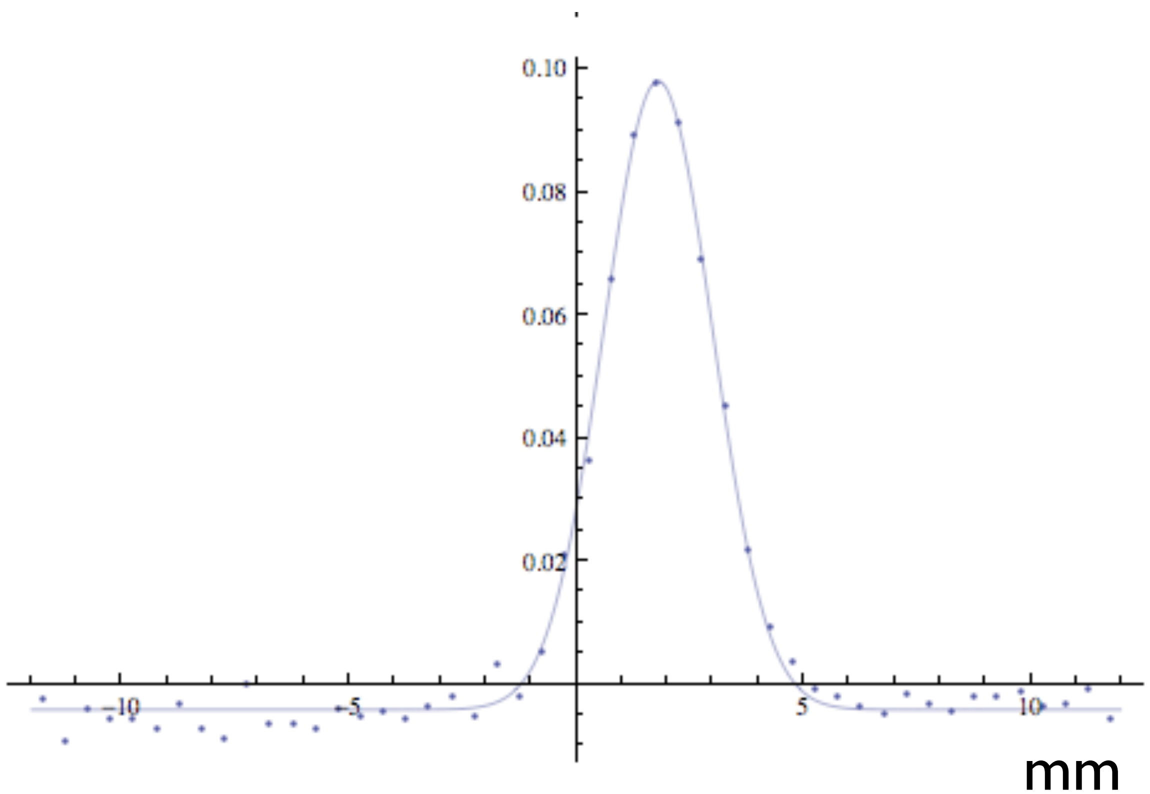
Beam Width at the Target

horizontal profile



	Estimate	Standard Error	t Statistic	P-Value
a	0.307776	0.00405775	75.8489	1.97904×10^{-47}
m	-0.00122003	0.015438	-0.0790278	0.937377
s	1.18545	0.016354	72.4872	1.36966×10^{-46}
c	-0.00327247	0.000333397	-9.81554	1.51435×10^{-12}

vertical profile

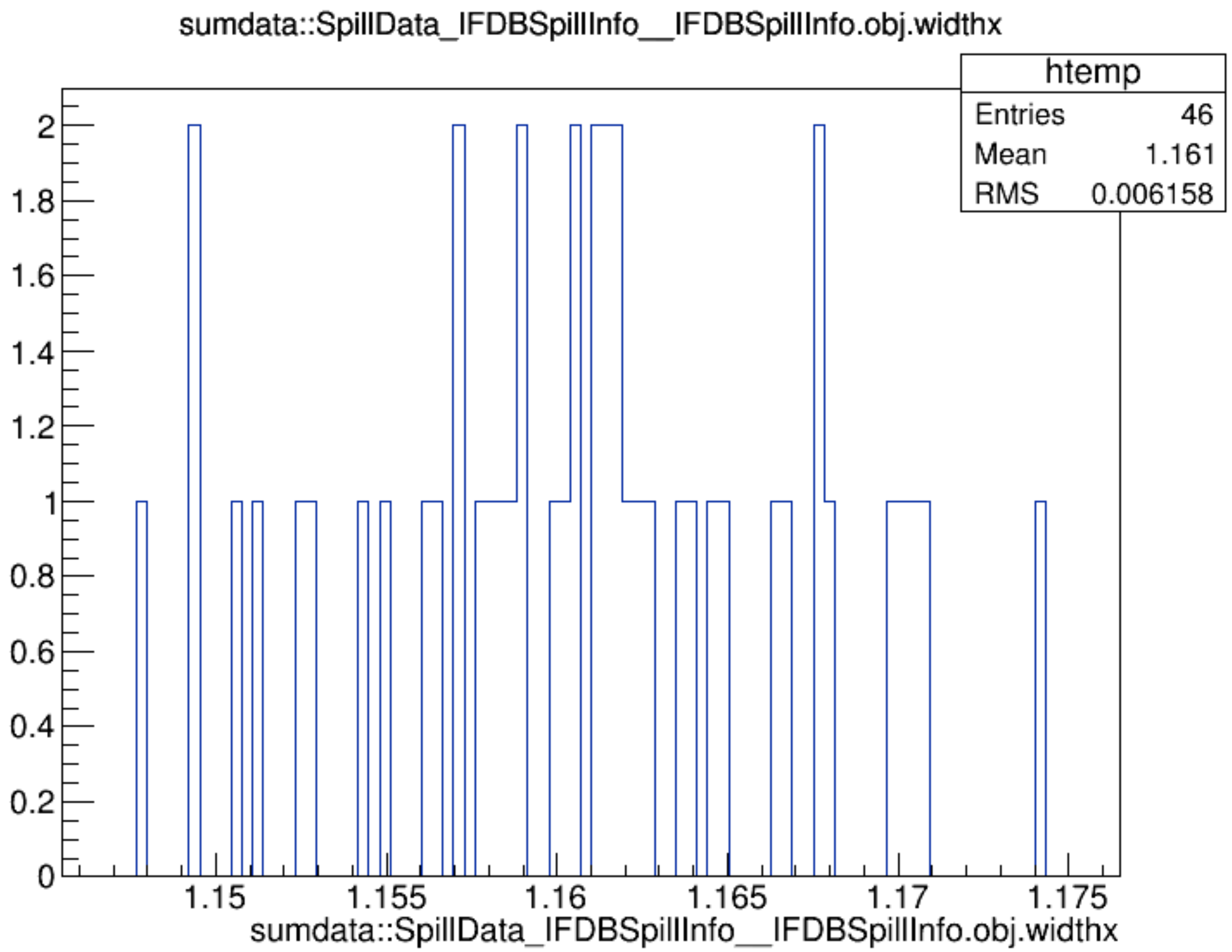


	Estimate	Standard Error	t Statistic	P-Value
a	0.305578	0.00480934	63.5384	3.75813×10^{-44}
m	1.80134	0.0185254	97.2362	4.83302×10^{-52}
s	1.19305	0.0196339	60.7647	2.50923×10^{-43}
c	-0.00418542	0.00039389	-10.6259	1.32435×10^{-13}

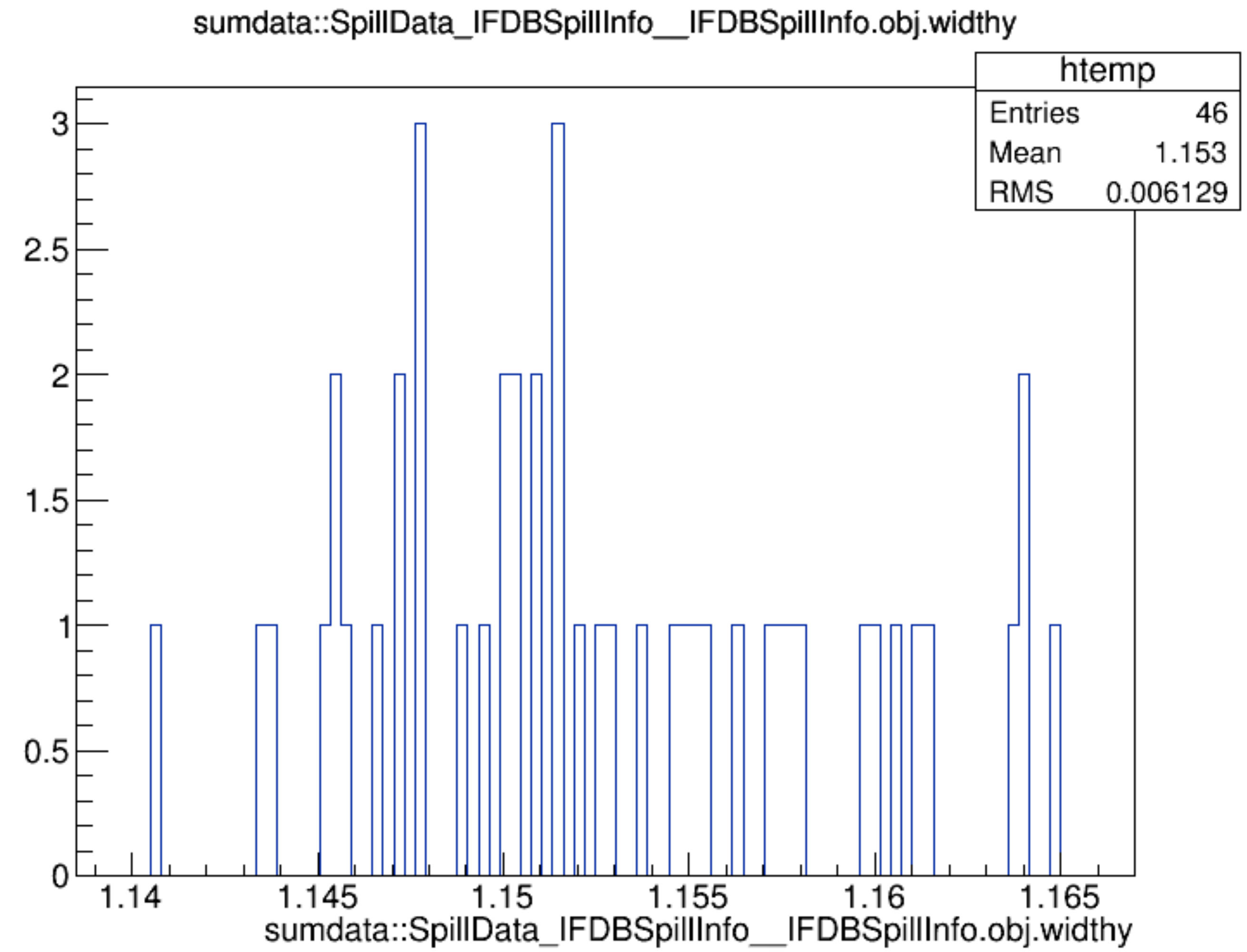
The quality of these fits is typical of the profile fits.

Beam Width at the Target

horizontal width



vertical width

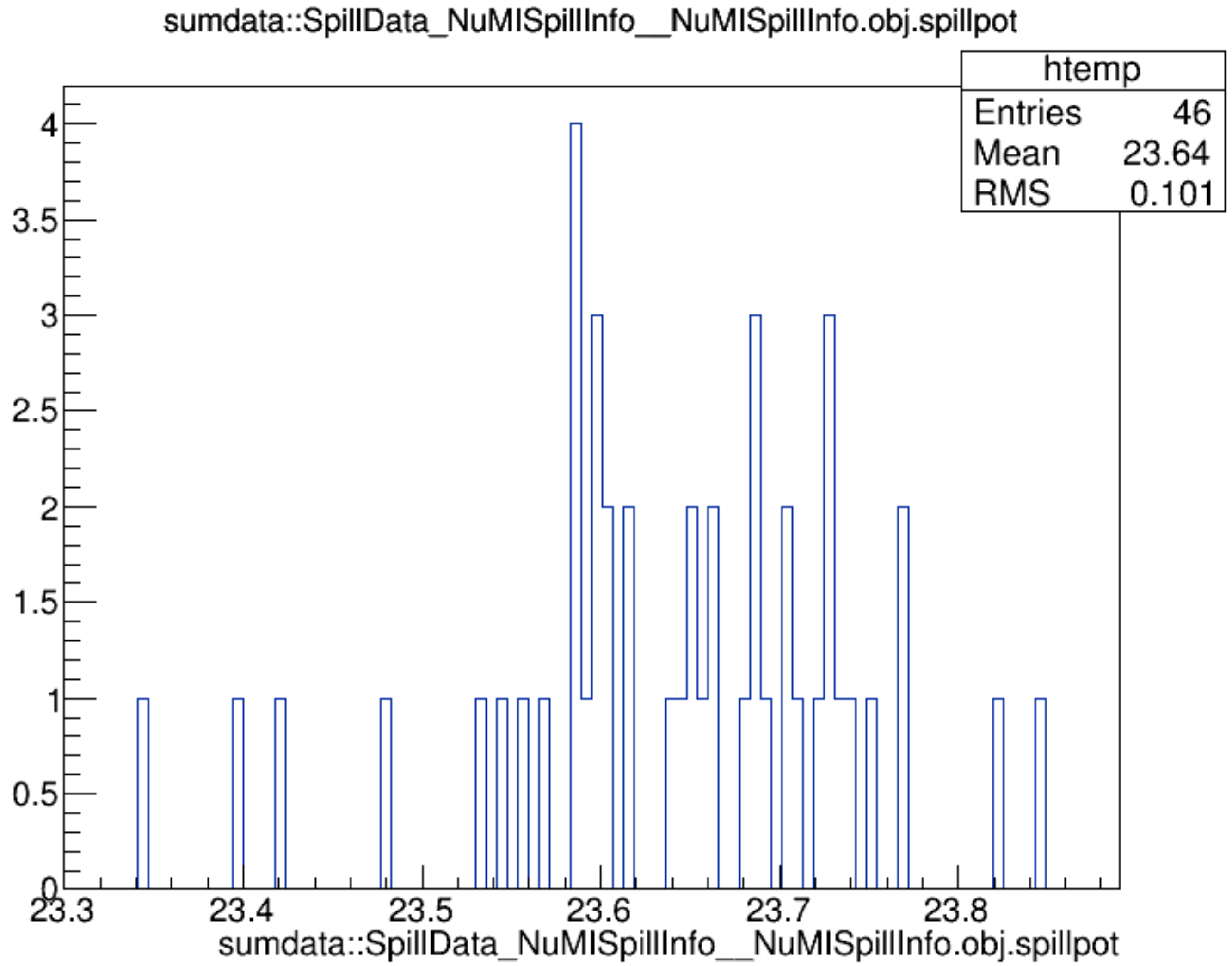


The horizontal and vertical widths of the beam should be within the range:

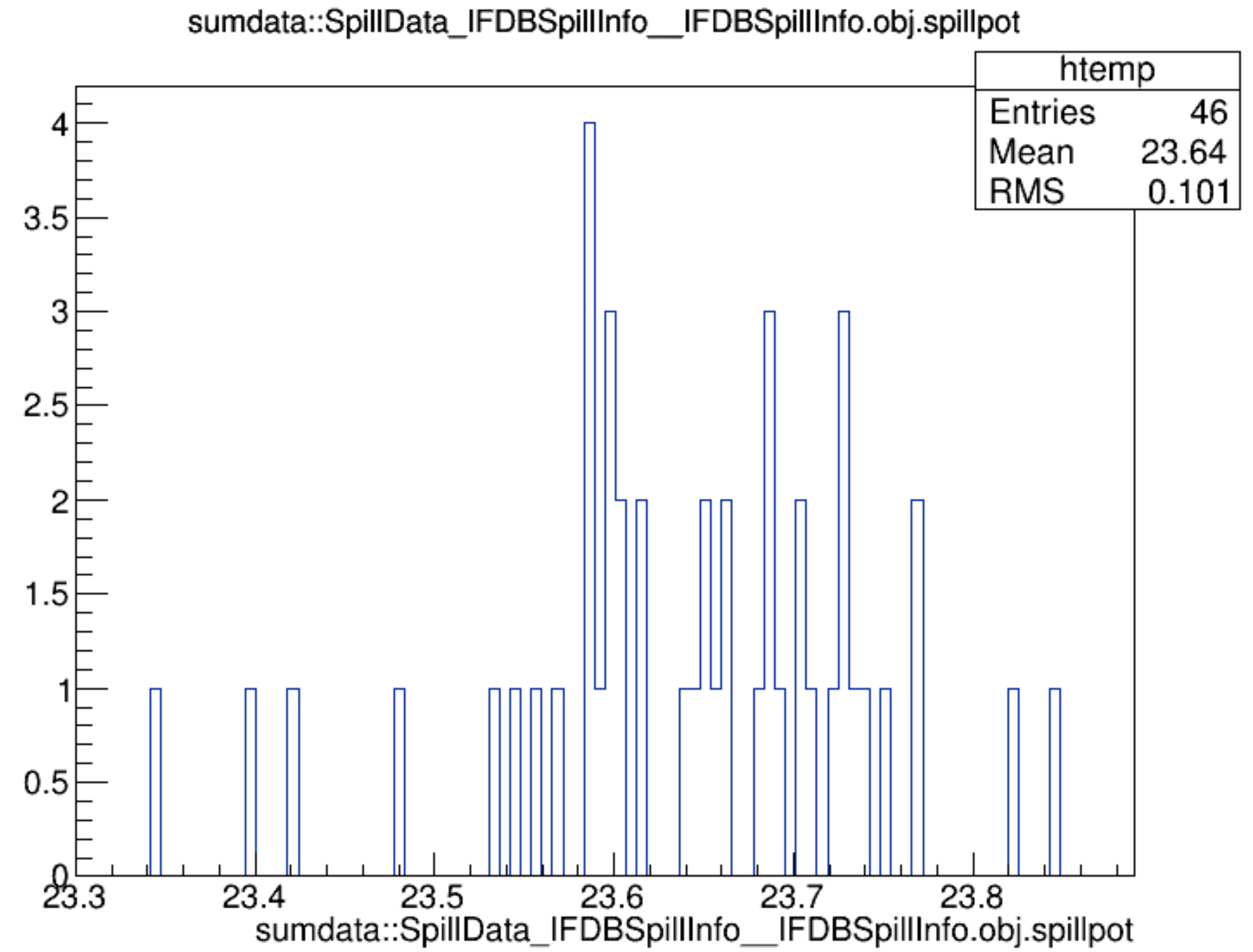
0.57 mm < widthx(y) < 1.58 mm

Sample POT Comparison Between IF Beam DB and NuMI Beam DB

NuMI



IFDB



This level of agreement for the POT on a spill-by-spill basis is typical.
(This subrun was not hand-picked.)

Past Issues

Getting to this point has not been without problems. Perhaps the two largest issues were the following.

- While doing validation work, I noticed some very strange behavior in the response of the IFDB to queries. The query results seemed to depend on (1.) what DB cache time was used and (2.) how the time for which data was requested was defined in the module. Additionally, I was getting data for times that were very far (in units of spill period) from the requested time.

- The performance of the database, once we started using the module in production work on the grid, was initially terrible. The database was being overwhelmed, so our jobs were getting hung up and often timing out.

Query Issues

EXACT TIME USED, 10 SEC CACHE

DAQ time:1386402791.478991

Exception: No data available for this time: 1.3864e+09

EVENT TIME USED, 10 SEC CACHE

evt time: 1386402791 sec, 478990937 nsec

DAQ time:1386402791.478991

Exception: No data available for this time: 1.3864e+09

EXACT TIME USED, 30 SEC CACHE

DAQ time:1386402791.478991

time in cache: 1386402803.342000
time in cache: 1386402815.343000
query interval: 1386402803.3420 to 1386402815.3430

closest time (sec): 1386402803
closest time (ns): 342000008
delta time (DAQ-DB): -11.8630 seconds
spill POT: 5.7061

EVENT TIME USED, 30 SEC CACHE

evt time: 1386402791 sec, 478990937 nsec

DAQ time:1386402791.478991

time in cache: 1386402778.467000
time in cache: 1386402790.467000
query interval: 1386402778.4670 to 1386402790.4670

closest time (sec): 1386402790
closest time (ns): 467000008
delta time (DAQ-DB): 1.0120 seconds
spill POT: 5.5877

inconsistent

Query Issues

EXACT TIME USED, 10 SEC CACHE

DAQ time:1386402791.478991

Exception: No data available for this time:
1.3864e+09

EVENT TIME USED, 10 SEC CACHE

evt time: 1386402791 sec, 478990937 nsec

DAQ time:1386402791.478991

Exception: No data available for this time:
1.3864e+09

this time should have been picked up by these queries

EXACT TIME USED, 30 SEC CACHE

DAQ time:1386402791.478991

time in cache: 1386402803.342000

time in cache: 1386402815.343000

query interval: 1386402803.3420 to 1386402815.3430

closest time (sec): 1386402803

closest time (ns): 342000008

delta time (DAQ-DB): -11.8630 seconds

spill POT: 5.7061

EVENT TIME USED, 30 SEC CACHE

evt time: 1386402791 sec, 478990937 nsec

DAQ time:1386402791.478991

time in cache: 1386402778.467000

time in cache: 1386402790.467000

query interval: 1386402778.4670 to 1386402790.4670

closest time (sec): 1386402790

closest time (ns): 467000008

delta time (DAQ-DB): 1.0120 seconds

spill POT: 5.5877

Query Issue Resolution

EXACT TIME USED, 10 SEC CACHE

DAQ time:1386402791.478991
time in cache: 1386402790.467000
time in cache: 1386402803.345000
query interval: 1386402790.4670 to 1386402803.3450
closest time (sec): 1386402790
closest time (ns): 467000008
delta time (DAQ-DB): 1.0120 seconds
spill POT: 5.5877

EVENT TIME USED, 10 SEC CACHE

evt time: 1386402791 sec, 478990937 nsec
DAQ time:1386402791.478991
time in cache: 1386402790.467000
time in cache: 1386402803.345000
query interval: 1386402790.4670 to 1386402803.3450
closest time (sec): 1386402790
closest time (ns): 467000008
delta time (DAQ-DB): 1.0120 seconds
spill POT: 5.5877

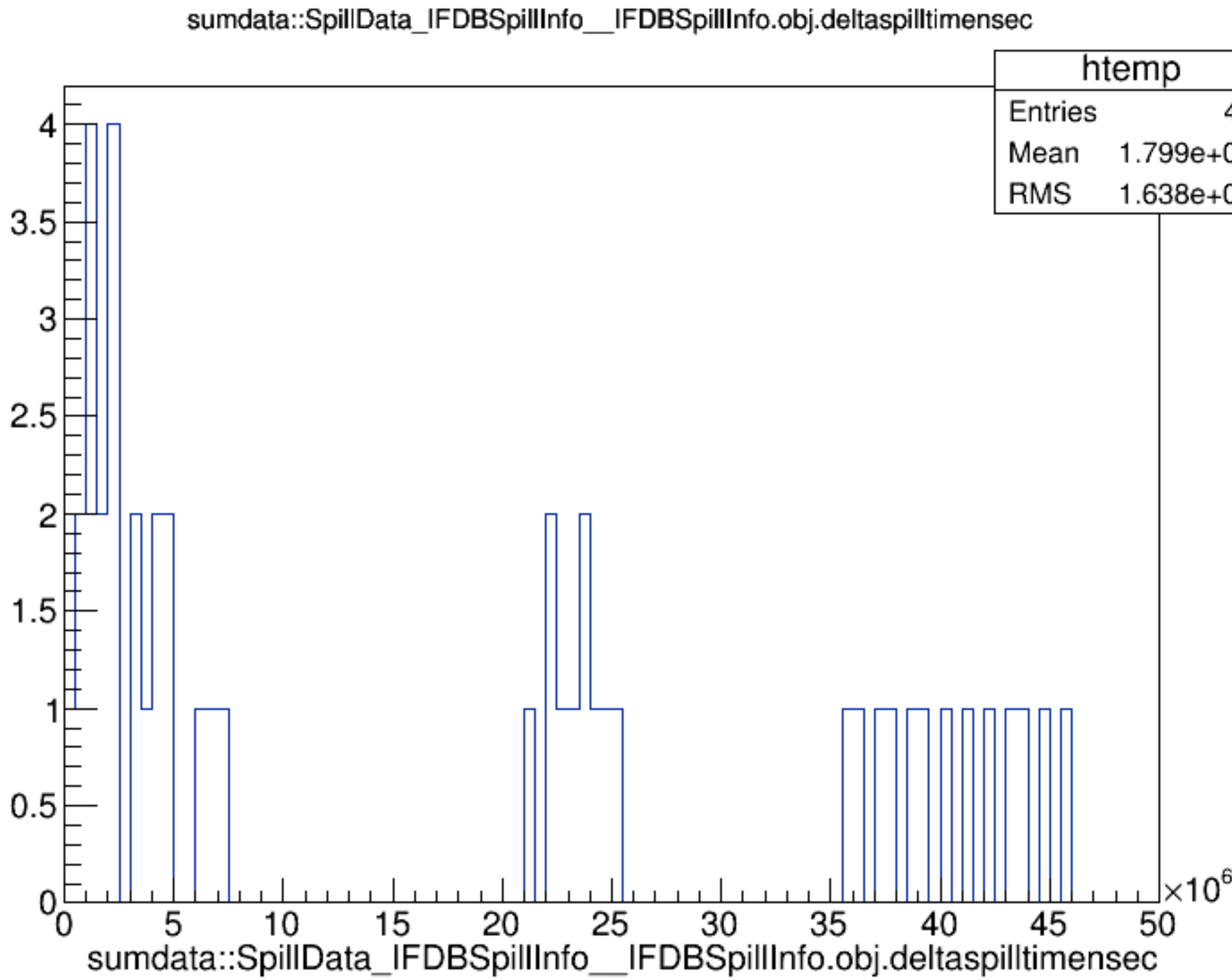
EXACT TIME USED, 30 SEC CACHE

DAQ time:1386402791.478991
time in cache: 1386402790.467000
time in cache: 1386402803.345000
time in cache: 1386402815.348000
time in cache: 1386402827.348000
query interval: 1386402790.4670 to 1386402827.3480
closest time (sec): 1386402790
closest time (ns): 467000008
delta time (DAQ-DB): 1.0120 seconds
spill POT: 5.5877

EVENT TIME USED, 30 SEC CACHE

evt time: 1386402791 sec, 478990937 nsec
DAQ time:1386402791.478991
time in cache: 1386402766.467000
time in cache: 1386402778.467000
time in cache: 1386402790.467000
time in cache: 1386402803.345000
query interval: 1386402766.4670 to 1386402803.3450
closest time (sec): 1386402790
closest time (ns): 467000008
delta time (DAQ-DB): 1.0120 seconds
spill POT: 5.5877

Time Difference Between Event Time and DB Time



Currently, the time difference cut is 1 x 10^9 nsec, which is the value the MINOS Collaboration used for their beam quality cuts. This value is rather large considering spills are currently ~1.67 seconds apart. The typical times differences are less than 50 ms, so the time difference cut may be modified in the future.

IFDB Peformance: Timing for DB Access

- The time the IFDB queries were taking per numi file was a concern to the Production Group.
- I started to collect some basic timing performance metrics.
- In addition to long query times, there was some strange behavior in the frequencies of the queries. For example, for a 30 second cache, the DB was re-caching data for every event, even though the events were separated by less than 30 seconds. (See left column on the next slide.)
- Igor and his team made many changes (I have documentation of these) to the DB, and the query times are now much reduced. The strange behavior in the frequencies of the queries also seems to be resolved.

IFDB Access Performance

PRIOR TO CHANGES

Times for first query for a given spill (milliseconds):

152, 452, 146, 116, 94, 119, 99, 93, 84, 220, 134, 111, 89, 189, 110, 164, 130, 184, 137, 104, 90, 178, 114, 121, 106, 183, 121, 110, 164, 207, 132, 96, 114, 200, 102, 131, 95, 129, 109, 89, 70, 150, 188, 137, 122, 240, 132, 116, 153, 142, 134, 132, 119, 191, 146, 101, 104, 204, 181, 99, 103, 1127, 243, 111, 106, 1456, 235, 117, 106, 317, 116, 95, 70, 203, 91, 111, 100, 320, 180, 129, 103, 186, 152, 121, 93, 153, 217, 130, 220, 170, 146, 115, 98, 168, 140, 114, 155, 148, 125, 103, 90, 134

(note: the first query takes 90% of the total)

Total times for all queries for a given spill (milliseconds):

164, 454, 148, 118, 97, 121, 101, 95, 222, 136, 113, 92, 193, 112, 167, 131, 186, 139, 106, 94, 180, 116, 123, 107, 185, 124, 112, 166, 208, 135, 98, 115, 202, 105, 133, 97, 131, 110, 90, 71, 191, 140, 124, 242, 133, 119, 155, 144, 136, 135, 119, 193, 148, 103, 106, 206, 183, 101, 105, 1129, 245, 114, 108, 1459, 235, 120, 109, 319, 117, 98, 71, 204, 113, 100, 323, 184, 131, 107, 190, 155, 123, 95, 155, 220, 130, 221, 172, 148, 116, 99, 170, 142, 117, 157, 148, 129, 103, 92, 138

Average time for all queries per spill: ~164 ms

AFTER CHANGES

Times for first query for a given spill (milliseconds):

246, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 271, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 231, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 208, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 170, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 289, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 305

(note: the first query takes 90% of the total)

Total times for all queries for a given spill (milliseconds):

256, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 275, 1, 1, 1, 9, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 236, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 213, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 174, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 294, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 309

Average time for all queries per spill: ~18 ms

Both tests were done with a 30 second cache.

Conclusions and Future Improvements

- The IFDBSpillInfo module accesses the IFDB to obtain beam monitoring device information. It calculates POT and determines the "goodness" of POT on a spill-by-spill basis using cuts on beam parameters.

- There are a few changes in the works:

  The position monitor data (from HITGT and VITGT) for dates prior to December 3rd need to be copied from the MINOS database to IFDB so that we can calculate intensity-weighted positions.

  Once the beam profile monitor at "121" is installed permanently, we will use extrapolation to calculate the beam width at the target.

  We will soon be adding the beam quality cut information into our OnMon or Nearline monitoring in our control room, so that we will have real-time information about the quality of the spills we are receiving.