

Beamspot Action Items

Current Action Items

Details below

1. Menu tests: setup additional chains and run them (*David+Ignacio*)
2. Detector configurations: check performance with individual Si layers off (*David*)
3. Pile-up validation: measure PV efficiency with multiple interactions (*David+Ignacio's scripts*)
4. Online vs. Offline beamspot comparison: compare both w.r.t. truth and w.r.t. offline (*David*)

Online Measurement

1. *Rainer+Philippe*: Check out the track impact parameter algorithm and run it in athena. Look through the histograms it produces and compare with the vertex algorithm.
2. *David*: Identify the set of histograms from the vertex algorithm that will be common with the impact-parameter algorithm. Encapsulate in a python class/method that can be executed from either algorithm.
3. *David*: Instantiate a second instance of the vertex algorithm with different parameters in the same job and make sure that the method produces two distinct sets. of histograms for both the algorithm-specific part as well as for the common part.
4. *David+Rainer*: Do a low-level check on how the gatherer handles statistics. Verify that the individual contributions add up correctly, including event weights and errors.
5. *Rainer*: Further discuss the pROS redistribution scheme with Benedetto, Per, Hans-Peter.
6. *David*: Measure performance with only the inner 1,2,3.. layers off in the SCT. Also, ask Andrea how to run SiTrack with requiring only 3 hits per track
7. *David+Ignacio*: Test different menu configurations
 - a. RND0_filled + FS_tracks: backup trigger for physics running
 - b. MBTX + FS_tracks: for van der Meer scans
 - c. Verify the actual configuration of the current unseeded FS_tracks menu (beamspot_vtx_fs_tracks)
8. *David+Ignacio*: Pile-up validation
 - a. Measure vertex finding efficiency in events with multiple interactions
 - i. 25ns, 450ns, no pile-up
 - b. Compare with offline vertices

Offline Validation

1. *David*: Compare online and offline measurements
 - a. Compare both online and offline with truth information
 - b. Compare online to offline event by event
 - c. Compare "beamspot": use full statistics for each
2. *David+Rainer*: Load our own online beamspot parameters for FDR2 and demonstrate how to access them from an offline job.

Completed or Closed Action Items

1. *David*: ~~Process the same (102) ranges of FDR2 data that Juerg produced offline beamspot measurements for with the online algorithm and compare. Find/understand differences. Compare alignment and other conditions. Compare resolutions and see if they make sense.~~
 - a. This has now resorted to running the online and offline beamspot on the same data (not the FDR2 due to DB problems)
2. *David+Rainer*: ~~Identify a tool to retrieve + load beamspot parameters in the conditions database. Extract the offline numbers that Juerg had loaded previously.~~
 - a. One pair that works: [Publishing with AtlCoolMerge](#) and browsing with [AtlCoolConsole](#)
3. *Philippe+David*: ~~Optimize track/cluster/vertex selection cuts, in particular for low luminosity phase.~~
4. *David*: ~~Identify the location for adding xml so that beamspot histograms are displayed in online monitoring.~~
 - a. /moncfg/beamspot on the P1 central file server
5. *Philippe+Rainer*: ~~Take the example of an existing xml file and tweak it to show some distributions from our vertex algorithm.~~
 - a. See: [OHPNexus User Guide](#)
6. *David+Rainer*: ~~Follow up with Tomasz on how the per lumi block histograms are stored, in particular, on a path with the correct run number and under the /EXPRESS folder.~~
7. *All*: ~~Play with the xml to see how much information can be extracted, e.g., mean, truncated mean, RMS, error on mean.~~
 - a. [The "Configurator" that lets one create xml files](#)
8. *Philippe*: ~~Contact Davide Caforio about DIP interface. Find out what else is needed to transmit beamspot parameters.~~
 - a. Instructions for publishing data are here: [Publishing via DIP](#)
9. *Rainer+Philippe*: ~~Create a PVSS project for the beam spot and understand how to store and retrieve extracted parameters.## Some information here: [Archiving in PVSS](#)~~
 - a. No more need for this, as the PVSS interface and project will be handled by Central DCS