tpass1 (A test pass before starting the pass1)

This is a test pass before starting the pass1

General info

- The hps-java tag is vCondtionTest
- JAR: /u/group/hps/hps_soft/hps-java/hps-distribution-3.11-ConditionTests-bin.jar
- JSub XML file
- All Steering files taken from JAR (i.e. they are used with the option -r)
- Logs: /lustre/expphy/work/hallb/hps/data/physrun2016/tpass1/logs
- The Detector is HPS-PhysicsRun2016-v5-3-fieldmap_globalAlign
- Only 10% of one run (7796) is processed
- files on work disk /work/hallb/hps/data/physrun2016/tpass1
- No file from this will go to tape

Outputs

- Recon.slcio
- DST.root
- ntuple moller
- ntuple fee
- ntuple trident

This test pass includes

- New geometry from Alessandra.
- event time stamp is fixed
- SVT bias and position flags expected to be fixed on this pass
- Track-cluster matching can be Re-processed on this files.
- Condition system, is choosing condition that are created most recently,
  instead of choosing the one that are modified most recently, this seems to reproduce pass0 reconstruction time.

Things that still needs to be fixed before going to pass1

- ~~Include Bad SVT channels~~ (Next pass)
- track parameters for detached vertexes (patched, but still needs rigorous solution)
- check all calibration are good (done)
- skim steering files are should be finalized (done)
- ~~Time dependent ECal gains~~ (Next pass)
- Check the SVT bias and positions flags before recon, and don't recon, if FLags are down (done)

Change to Java8

Recon with Java 8 is faster, instead of 15 hours, now recon is done in about 11 hours, so we will use Java8 instead of java7

asking 2.4 GB RAM from Auger instead of 4 GB, didn't cause any problem, for future we will use 2.4GB RAM, and 1.4GB heap size

Where's the target?

Please see these slides. There seems to be some agreement between FEE, Møllers and V0s for a target position at ~ -1mm.

I would be OK with leaving it at z=0 for this pass.

Issues

- Failed partitions: Several partitions timed out during reconstruction
- Memory Leak: There appears to be a memory leak in our current reconstruction. Solving this would presumable allow us to run our jobs on a larger number of machines at JLab without requiring the large memory footprint we now do.