

OBF and Configuration - Getting to the Finish Line

A final list of items OBF-related that need doing.

What	Who	Status
MootSvc	Joanne	DONE using MootSvc v1r1p1 (Eric's requests concerning filter configuration data are included) in GR v15r7 and beyond - need to make sure the JO parameters are handled correctly in basicOptions and in pipeline the Gleam unit test will use MOOT - but the Gleam/src/basicOptions.txt (as of v7r0p2) is set up NOT to use MOOT via MootSvc. noMoot = true; which probably means we need to also set OnboardFilter.UseMootConfig = "false"; in basicOptions, but set it to true in default.txt for the unit test. Meanwhile, the pipeline obviously will run with MOOT. ** see below for OnboardFilter JO commentary from Tracy
Provide TDS location for MOOT Master Key and PrescaleFactor	Heather	Requested by Eric C. DONE - in GR HEAD1.1109 to be included in tag v15r8 and beyond. Bryson has promised to provide the moot key via CHS/eventFile - but it is not clear when that will actually be available. See JIRAs: ICS-770 and LONE-62
Provide method to merge state and prescaler info into one quantity	Heather	Requested by Eric C.
Fill "real" filter bits in digis when running MC	Heather	Starting to look at that now.
Repackaged Linux OBF external library	Navid	Passed to Super-Navid for completion- and is now completed - included in GR HEAD1.1112
New merit variables for 'real' filter bits	Leon /Heather	Was waiting for fix up of AnaTup to remove use of FluxSvc (Toby) and clean up of GPS singleton, which is now complete. After hearing from Anders, due to our desire to fill the "real" filter bits when running MC - we will store the status words for all filters (even though HIP, DGN, PASS_THRU will not populate it for real data). What other quantities should we store, state? anything else? We have a new AnaTup that's been cleaned up v2r41p3 - now to get the hardware filter bits in: to include status word, state, and prescaler for all filters (despite that status word only exists for MIP and GAMMA, and prescaler is not available as of version 0 of the LpaHandlers)
Document digiRootData's new entries for filter bits	Heather	Will update existing Confluence page then migrate to Doxygen in the workbook with Chuck's help
Make a new GR	Heather	a new one every day it seems

***Stolen from an email from Tracy:*

An arbitrary default setting of parameters as I type this is:

- OnboardFilter (the algorithm) - just the important ones*# "UseMootConfig" - set to false by default, true means get configuration info from Mootb)
 - "RejectEvents" - set to false by default, true means OnboardFilter will use "active" filters to accept/reject eventsc)
 - "FailNoEbfData" - set to false by default, true means OnboardFilter "crashes" (gracefully) if ebf data is missing
 - "FilterList" - this is a list of filters to configure and run, the default is Gamma, MIP, HIP, DGN and
 - "FilterTrack" (grabs track info for GRB)The rest are highly specialized and don't matter.

Running the pipeline will, at this time, require:OnboardFilter.useMootConfig = true;

- GammaFilterTool - this controls the Gamma Filter*# "LeakAllEvents" - set to true by default, causes the Gamma Filter to report on all events which means it runs all stages of the filterb)
 - "Configuration" - set to "" by default, if set this will override the configuration in the Master config file (and, Moot too I think)
 - "GamFilterMask" - set to 0 by default, if set this will mask out veto bits during processing - SHOULD NOT BE USED BY NOVICES
 - There are also a collection of JO parameters to allow you to change the parameters of the Gamma Filter, these should also NOT BE USED
- All of the other filters (DGNFilterTool, HIPFilterTool, MIPFilterTool) also have a "LeakAllEvents" parameter which is set to true. They don't have any other JO parameters at this time, but it will not be much work to be able to provide the same mechanism the Gamma Filter has for changing their parameters.
As it stands right now, if "UseMootConfig" is set to true, then OnboardFilter will look for MootSvc (which must be included by the controlling JO files). If it cannot find MootSvc then it will "crash". If MootSvc is found then all tools will use it to do the configuration. And, filter initialization will occur on the first event so that Moot can look up what it needs once the first data is read in.

Note: Moot only contains information for "active" filters which JJ tells me are those that participate in the decision to accept/reject events.

At the same time, we may also run, for example, the MIP filter but will have no configuration information for it from Moot... in this case it will run "normal" mode and use the configuration defined in the master configuration file.If UseMootConfig is "false" then the filters will initialize during the initialize method of OnboardFilter. They will configure according to the master configuration files for each filter and will run in "normal" mode. This can be overridden by JO parameters for the Gamma filter now, and I'll add that for the other filters straight away.Finally, each event OnboardFilter will look for the MetaEvent TDS class. If found then it will check the run mode and see if it has changed from the currently cached value (I wish there was a better way!). If it has changed then it will change modes for all "active" filters only.Ok, let me know if defaults are ok or any other issues that might be obvious from what I wrote (or didn't write).

Thursday June 19, 2008

List of tags to add to GR HEAD:

Trigger v6r3p1
OnboardFilterTds v0r8p1
AnalysisNtuple v2r43p1
MootSvc v1r1p3
ConfigSvc v0r2p4
Gleam v7r0p7