

art - Feature #4916

Implement SelectRuns and SelectSubRuns

11/06/2013 01:57 PM - Christopher Backhouse

Status:	Feedback	Start date:	11/06/2013
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Event Loop	Estimated time:	0.00 hour
Target version:		Spent time:	0.00 hour
Scope:	Internal	SSI Package:	art
Experiment:	NOvA		
Description			
As options on RootOutput, with behaviour analogous to SelectEvents.			
Spun off from Issue 4895			
Related issues:			
Related to art - Support #4895: No way to filter runs/subruns out of output file		Closed	11/01/2013
Related to art - Support #10545: Returning false from EDFilter::beginSubRun()...		Closed	10/15/2015

History

#1 - 11/11/2013 11:31 AM - Christopher Green

- Category set to Event Loop
- Status changed from New to Feedback
- Experiment NOvA added
- Experiment deleted (-)
- SSI Package art added
- SSI Package deleted ()

Would it be acceptable to offer explicit options to the RootOutput module to:

- Omit subrun records for which no events have been written; and
- Omit run records for which no subrun records have been written?

The more selectSubRuns and selectRuns options you suggest would be harder to implement, and open the possibility of a mis-configured job being able to write events with no run or subrun records into a file, which we believe would qualify currently as a corrupted (and therefore unreadable) file.

#2 - 11/11/2013 01:18 PM - Christopher Backhouse

I'm not quite sure this works. I want to be able to have eg retain a subrun with products in it, but all of the constituent events filtered out (which is currently possible). And likewise with a run, which isn't currently possible.

A worse failure case I guess is that with my scheme one could try and cut out a subrun that had events still remaining in it. But presumably it should be possible to flag these as errors as soon as the module makes such an inconsistent determination.

You could make the action of Select[Sub]Runs dependent on this criterion that nothing from the lower level has been written. That seems like a good way to retain safety.

#3 - 04/17/2015 10:24 AM - Kyle Knoepfel

- Target version set to 521

#4 - 10/15/2015 08:02 PM - Christopher Backhouse

I had a feeling this feature, or one similar to it, had been implemented, but I can't find any evidence in redmine.

I've run into the need for this again. I'm making skim files of a few events out of many hundreds of runs. I'd prefer my output file to be as minimal as possible, only consisting the runs that are actually needed to contain the events I select.

#5 - 10/16/2015 08:56 AM - Kyle Knoepfel

Issue [#4998](#) was implemented as a mitigation to the problem you have discussed. However, as you can tell from the status of this particular feature, it has not been implemented.

#6 - 10/16/2015 01:27 PM - Christopher Backhouse

I'm not sure [#4998](#) covers this case. That's for unconditional dropping of all events in a subrun or all subruns in a run. Here I want to be able to make the decision on a case-by-case basis at runtime.

#7 - 11/02/2015 11:51 AM - Christopher Green

- *Related to Support #10545: Returning false from EDFilter::beginSubRun() does not skip the events added*

#8 - 10/23/2017 12:12 PM - Kyle Knoepfel

- *Target version deleted (521)*