art - Bug #25117

event mixing : strong nonlinear dependence of job timing on the "instantaneous luminosity"

10/25/2020 05:30 PM - Pavel Murat

Status:	Closed	**Start date:**	10/25/2020
Priority:	Normal	**Due date:**	
Assignee:	Kyle Knoepfel	**% Done:**	100%
Category:	Infrastructure	**Estimated time:**	8.00 hours
Target version:	3.05.02	**Spent time:**	7.00 hours
Occurs In:	3.05.01	**Experiment:**	Mu2e
Scope:	Internal	**SSI Package:**	art

Description

Dear art developers,

I'm observing a really slow and very non-linear, vs the "instantaneous luminosity", or a number of mixed-in particles, performance of the art mixing jobs.
For a standard Mu2e mixing setup, the mean time per event depends on the number of input (mixed-in) particles as ~ N^2 and for the proton pulse intensity
of 12e7, the highest simulated one, for an executable compiled in optimized mode could reach more than 15 minutes per event - see attached plot.

As mixing is a linear superposition of the input particles and their hits, one wouldn't expect the quadratic term in time/event = a + b*N +C*N^2 to be significant, however it is.

Present level of mixing job performance has a significant impact on the dataset production for the '2020 Mu2e sensitivity update, it would be really helpful if experts could
take a look, and the release v3_05_01 which we are using was patched.

To reproduce the performance problem one could login into one of the Mu2e interactive platforms and do the following:

```
source /cvmfs/mu2e.opensciencegrid.org/setupmu2e-art.sh
source /mu2e/app/users/murat/su2020_prof/setup.sh
mu2e -c /mu2e/app/users/murat/su2020_prof/su2020/mnbs0/s4_no_primary1_mnbs0.fcl -n 100
```

Note: the very first event has very low simulated pulse intensity, so it is not characteristic, simulation of event # 97, however, takes more than 15 min+

-- many thanks, regards, Pasha

Associated revisions

Revision 07e5d18f - 11/13/2020 11:16 AM - Kyle Knoepfel

Resolve issue #25117: optimize map_vector concaatenation.

History

#1 - 10/26/2020 10:25 AM - Kyle Knoepfel

- Estimated time set to 8.00 h

- Assignee set to Kyle Knoepfel

- Status changed from New to Assigned

- Tracker changed from Bug to Support

We will run some profiling on this workflow and attempt to ascertain whether this is expected behavior based on the chosen mixing algorithm, or whether we can improve the mixing procedure.

#2 - 11/12/2020 08:16 AM - Kyle Knoepfel

- Status changed from Assigned to Feedback

I get the following error when trying to execute the instructions above:

```
-bash-4.2$ map --profile --start --nompi $(type -p mu2e) -c /mu2e/app/users/murat/su2020_prof/su2020/mnbs0/s4_
no_primary1_mnbs0.fcl -n 100
Arm Forge 20.0.3 - Arm MAP

Profiling          : /cvmfs/mu2e.opensciencegrid.org/artexternals/art/v3_05_01/slf7.x86_64.e19.prof/bin/mu2e -
c /mu2e/app/users/murat/su2020_prof/su2020/mnbs0/s4_no_primary1_mnbs0.fcl -n 100
Allinea sampler    : not preloading
MPI implementation : Auto-Detect (None)

Failed to parse the configuration file '/mu2e/app/users/murat/su2020_prof/su2020/mnbs0/s4_no_primary1_mnbs0.fc
l' with exception
---- Parse error BEGIN
  Local lookup error
  ---- Can't find key BEGIN
    BLIND_TIME (at part "BLIND_TIME")
  ---- Can't find key END
  at line 451, character 64, of file "/mu2e/app/users/murat/su2020_prof/JobConfig/common/su2020_templates.fcl"

  included from line 41 of file "/mu2e/app/users/murat/su2020_prof/JobConfig/common/su20201.fcl"
  included from line 11 of file "/mu2e/app/users/murat/su2020_prof/su2020/mnbs0/s4_no_primary1_mnbs0.fcl"

  services.ProditionsService.strawElectronics.flashEnd         : @local::BLIND_TIME
                                                                  ^
---- Parse error END

Art has completed and will exit with status 90.
```

#3 - 11/12/2020 04:51 PM - Kyle Knoepfel

The problem is understood. To concatenate cet::map_vector objects during product mixing, it is necessary to adjust the keys so that elements of the same key are not discarded. Although this key adjustment is done correctly, the concatenation step unnecessarily calls a merge operation to ensure a sorted final collection. By construction, the concatenated collections will have disjoint keys, and it is already the user's responsibility to ensure an ordered collection. Simply appending the new key-adjusted elements at the end of the collection is therefore sufficient.

The below table shows the wallclock time for the above job using the current merging method and the proposed appending method.

Concatenation method	**100 events**	**Event 1:0:97**	
Merging (art 3.05.01)	3180 sec	1256 sec	
Appending (proposed)	189 sec	15 sec	*yes, the*

This will require a new version of art. Can Mu2e please tell us whether an art 3.05 or 3.06 bug fix release is necessary, or whether it is okay with waiting for art 3.07 in a couple weeks?

#4 - 11/12/2020 04:51 PM - Kyle Knoepfel

- % Done changed from 0 to 80

#5 - 11/12/2020 04:57 PM - Rob Kutschke

Thanks Kyle. I wil check with interested parties and get back to you.

#6 - 11/12/2020 10:54 PM - Rob Kutschke

Please produce the next bug fix release in the v3.05 series. Pasha et al are working from a stable older code that is still using v3.05. Our master branch is at v3.06.03 but we can wait for v3.07.x for that - we do not anticipate major production on the time scale of a few weeks.

#7 - 11/13/2020 11:21 AM - Kyle Knoepfel

- % Done changed from 80 to 100

- Target version set to 3.05.02

- Status changed from Feedback to Resolved

- Category set to Infrastructure

- Tracker changed from Support to Bug

- Occurs In 3.05.01 added

- SSI Package art added

- Experiment Mu2e added

- Experiment deleted (-)

Resolved with commits:

- cetlib:fbba024
- art:07e5d18

#8 - 11/18/2020 12:42 PM - Kyle Knoepfel

- Status changed from Resolved to Closed

Files

time_per_event_vs_inst_lum_1000ev.png	38.6 KB	10/25/2020	Pavel Murat