

## artdaq - Bug #5980

### The Aggregator runs slower when event grouping is enabled

04/21/2014 11:22 AM - Kurt Biery

<b>Status:</b>	New	<b>Start date:</b>	04/21/2014
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>	Under Consideration	<b>Estimated time:</b>	60.00 hours
<b>Target version:</b>		<b>Co-Assignees:</b>	
<b>Experiment:</b>			

#### Description

Last September, we discovered that the feature that allows us to group several consecutive events into a bunch in the EventStore that runs in the Aggregator causes noticeable performance issues. (The reason for grouping consecutive events together is to allow online monitoring to do event-to-event analyses.)

Here are some notes from September:

#### email from me

I was able to improve system performance by configuring the Aggregator so that it does not group  $N > 1$  events together before passing them to art. Some performance results are shown in the following table - these numbers have large error bars, but hopefully they show that an improvement has been made.

maximum event rate	Aggregator grouping of 4 events	no Aggregator grouping of events
10 usec gate width	345 events/sec	375 events/sec
100 usec gate width	87 events/sec	180 events/sec
300 usec gate width	34 events/sec	64 events/sec
500 usec gate width	20 events/sec	43 events/sec

These results were achieved with a system with 6 boardreaders (5 1720s, one 1495), 16 eventbuilders, and two aggregators. Disk writing and online monitoring were turned off in these tests.

I haven't looked into why the grouping of  $N > 1$  events could be slowing things down, but I see a call to the sort function on line 110 of NetMonTransportService\_service.cc that might be a place to start looking.

#### email from Mike Wang on the subject of reproducing the problem at the FNAL WH14 teststand

Using -g 600 -p 2000, here are the event rates for the first six samples printed to the console for a grouping of 4 and 1:

grouping = 4 (events/sec)

26.0122  
25.6448  
27.8478  
29.3477  
31.9142  
30.831

grouping = 1 (events/sec)

33.5078  
34.1949  
34.477  
34.864  
34.9473  
34.7972

#### History

#1 - 03/04/2016 08:54 AM - Kurt Biery

I couldn't reproduce this in v2\_05\_02 of the artdaq\_demo running on mu2eddaq01 (2x2x2 system). I increased the size of the data generated by the ToyFragment\_generators to get events around 3 MB. Maybe the time needed to generate the data washes out the problem?

**#2 - 01/13/2017 04:37 PM - Eric Flumerfelt**

- *Category set to Under Consideration*

- *Target version deleted (575)*