Subject: Re: FairWriteoutBuffer::FillNewData and object ownership (memory leak)

Posted by Mohammad Al-Turany on Sat, 26 Jan 2013 15:33:32 GMT

View Forum Message <> Reply to Message

Hi,

Sorry for the delay in answering this mail, in any case objects added to TClonesArray are added via new with placement, so you do not call the new and delete in the traditional way (That is why it is much faster than STL or TObjectArray). If the objects added to the TClonesArray do not allocate memory, it is enough to call the clear of the array to free the memory, other wise (e.g: your object has a TString or any other object inside it) then you need to call delete.

Normally each Task call the delete (Clear) of the TClonesArray in the finish event method.

Now for the Time based simulation it is more complicated because we use the TClonesArray::AbsorbObjects method which simply move objects from one array to the other, and it really move and not copy them. Now after moving and writing the objects a method FairRootManager::DeleteOldWriteoutBufferData is called to free the memory in the Buffer. In the Task the memory is freed by the Finishevent. in Other words, we have three TClonesArray:

1. The one used internally to read objects from tree
2. The one in the Buffer
3. The one in the task, which is connected to the output tree

Objects are moved from one array to the other by calling TClonesArray::AbsorbObjects, we clean the buffer from the rest and the Task has to clean what it write to the output tree.

regards,

Mohammad