Subject: Re: FairWriteoutBuffer::FillNewData and object ownership (memory leak)
Posted by Philipp Mahlberg on Mon, 25 Feb 2013 15:46:22 GMT
View Forum Message <> Reply to Message

Two additional notes concerning the copy creation and deletion of the FairTimeStamp objects in the timebased simulation.

1. From my point of view, the default FairTimeStamp::Modify function still causes a memory leak, as the newly incoming and further on ignored object is not deleted.

2. I am working on a kind of extended timebased simulation for the emc, where I pass interim objects (keeping track of all the needed information) to the event mixing buffers and construct a corresponding Emc waveform right before filling the data to the TClonesArray.

Therefore the interim object contains a pointer to the waveform generator, which will take care of the waveform building process. Since the TObject::Clone() copies the object via streaming, ROOT is constructing new generator objects etc. as well. A simple copy of the address would be sufficient, but the method here creates lots of (useless) objects and brings up a runtime error (ROOT complains about a missing default constructor of an abstract class involved). The latter should be fixed, but I think it is not the main problem though)

As mentioned before, the objects represent an interim state and are not primarily designed to be written into a file, what is done with the final waveforms.

At the moment, I refrain from getting an extra copy of the object by overriding the FillNewData Method.
One could also think of writing a customary streamer which then hopefully just copies the reference address stored in that specific pointer (I am not an expert on this at all, so I don't know weather it will work). But then I could never make use of all of ROOT streaming power to write an object with all its dependencies to file, e.g. for debug reasons. Perhaps it is also possible to distinguish between writing to a file and cloning the object in the streaming function?

Maybe somebody even knows a smarter way how to overcome the problem....