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

Posted by [Oliver Merle](#) on Sat, 26 Jan 2013 13:31:37 GMT

[View Forum Message](#) <> [Reply to Message](#)

Another point I do not quite understand is why what you call "active time" is fixed to -1 in the call to Modify (at FairWriteoutBuffer.cxx:186)

```
std::vector<std::pair<double, FairTimeStamp*> > modifiedData = Modify(std::pair<double, FairTimeStamp*>(currentdeadtime, oldData), std::pair<double, FairTimeStamp*>(-1, data));
```

In case I do not combine the signals given by both arguments but just use a dead time (aka active time) to emulate occupancy, I need the active time of the new hit in Modify - but it is replaced by -1.

Seems fishy to me, but maybe I just don't understand the deeper meaning of this mysterious number. Looking at the code of the SDS package doesn't clarify things either:

PndSdsDigiStripWriteoutBuffer.cxx:32

```
std::vector<std::pair<double, PndSdsDigiStrip*> >
PndSdsDigiStripWriteoutBuffer::Modify(std::pair<double, PndSdsDigiStrip*> oldData,
std::pair<double, PndSdsDigiStrip*> newData)
{
    std::vector<std::pair<double, PndSdsDigiStrip*> > result;
    std::pair<double, PndSdsDigiStrip*> singleResult;
    if (newData.first > 0)
        singleResult.first = oldData.first + newData.first;
    singleResult.second = oldData.second;
```

The if statement makes no sense because newData.first is always -1 as shown above (I don't find any other call to Modify either). The same if statement can be found in PndSdsDigiPixelWriteoutBuffer.cxx:31 and in the Gem package.

Seems to be inconsistent - what do I miss here?
