Subject: How to create more than one output event in FairTask Exec()-function?
Posted by C. A. Douma on Tue, 26 Mar 2019 11:41:36 GMT

Dear all,

I would like to design an R3BRoot FairTask that merges several .root-files.

So suppose that I create a TClonesArray* fOutput;
that I write its content with FairRootManaber::Register to an output-file,
and that I expand the fOutput-array with one object at a time
using the new-operator (inside the Exec()-function),

is there a way to manually call a command for writing fOutput
to the file and emptying it halfway inside the Exec()-function?
So that I essentially generate multiple events in the output-tree
for a single Exec()-call?

Many thanks in advance!
Christiaan Douma.

Subject: Re: How to create more than one output event in FairTask
Exec()-function?
Posted by Jan Mayer on Wed, 27 Mar 2019 09:34:57 GMT

Hello,

what is it you actually want to accomplish? What should be "merged" and why?

Quote:is there a way to manually call a command for writing fOutput
to the file and emptying it halfway inside the Exec()-function?
So that I essentially generate multiple events in the output-tree
for a single Exec()-call?
I do not think it is a good Idea to work against one of the main principles of the data processing
system. In- and Output files should have a 1:1 relationship - so one can use data from several
root files together and be sure its the correct event each time (see 14.16.2
https://root.cern.ch/root/htmldoc/guides/users-guide/Trees.html#example-
3-adding-friends-to-trees).

I suspect your problem is somewhere else, and can be solved differently.

Jan

Subject: Re: How to create more than one output event in FairTask

Exec()-function?
Posted by C. A. Douma on Wed, 27 Mar 2019 11:32:36 GMT

View Forum Message <> Reply to Message

Dear Jan,

The ultimate goal of this exercise is to do my simulations in multi-thread mode.
To avoid the complicated C++ multithread programming, my thoughts went to using
a bash-script for running, say, 4 copies of r3b_sim.C in parallel (in different shells).
Then, I can run your digitizer afterwards (also 4x in different shells).

However, this will then produce 4 distinct .root-files, all with their own
digitizer output tree. Hence, I wanted to write another FairTask that does
nothing more then merging these 4 files into one. But since FairTask
runs Exec() ones per event and reads & writes one event at a time,
taking event 1 from 4 different files & writing them as event 1,2,3,4
does not seem possible withing Exec(). Hence my question: can I manually
call a write & reset function for the output TClonesArray?

In that Case, I can use AddFriend() on the 4 digitizer files, read one
event from each file during one Exec()-call and then call the write & reset
function 4 times to produce 4 output events within that same Exec()-call.

Christiaan.

Subject: Re: How to create more than one output event in FairTask Exec()-function?
Posted by Jan Mayer on Wed, 27 Mar 2019 12:16:01 GMT

View Forum Message <> Reply to Message

Quote:This will then produce 4 distinct .root-files, all with their own digitizer output tree. Hence, I wanted to write another FairTask that does nothing more then merging these 4 files into one.

Aha! Now we're talking.

Quote: Hence my question: can I manually call a write & reset function for the output TClonesArray?

You don't need to do that. At all.

I mean, if you think about it, this must be such a common problem! Everyone would face that at some point, right? Even the experimental data is split into multiple files automatically. Since decades, people would have to work with that, right? Having their code run over terabytes of data in hundreds of root files!

There are two main solutions: Add the file contents together to form a new large file with

"hadd" (not recommended), or simply tell ROOT that the input is split over multiple files - in ROOT terms, that is called a "TChain".

The FairRoot equivalent / abstraction is in "FairFileSource", which has a function "AddFile", which does what it says.

/**Add ROOT file to input, the file will be chained to already added files*/
void AddFile(TString FileName);

Subject: Re: How to create more than one output event in FairTask Exec()-function?
Posted by C. A. Douma on Wed, 27 Mar 2019 13:16:02 GMT
View Forum Message <> Reply to Message

FairFileSource::AddFile() looks promising. How can I use it?

In a steering macro, I do

FairRunAna* run = new FairRunAna();
run->SetInputFile("<path to>/DigiFile.root");

Can I then just do
run->AddFile("<path to>/DigiFile_2.root");
and do run->Run(0,twice as many events as in a single file);

Christiaan.

Subject: Re: How to create more than one output event in FairTask Exec()-function?
Posted by Jan Mayer on Wed, 27 Mar 2019 13:27:26 GMT
View Forum Message <> Reply to Message

auto source = new FairFileSource("<path to>/DigiFile.root");
source->AddFile("<path to>/DigiFile_2.root");

auto run = new FairRunAna();
run->SetSource(source);

"SetInputFile" is deprecated anyway.

Also, with run->Run(0, 0); all available events will be processed. No need to specifying the exact number.

Subject: Re: How to create more than one output event in FairTask Exec()-function?
Posted by C. A. Douma on Wed, 27 Mar 2019 14:35:39 GMT

Thank you. I will try it.

Christiaan.