Subject: Runtime DB
Posted by Tobias Stockmanns on Tue, 29 May 2007 08:48:36 GMT
View Forum Message <> Reply to Message

Hi Simulators,

is somebody using the runtime DB to store and retrieve simulation parameters? I need some informations how to use it.

Cheers,

Tobias

Subject: Re: Runtime DB
Posted by StefanoSpataro on Tue, 29 May 2007 09:06:54 GMT
View Forum Message <> Reply to Message

Don't look at me...

Subject: Re: Runtime DB
Posted by Jens Sören Lange on Tue, 29 May 2007 14:43:29 GMT
View Forum Message <> Reply to Message

Hi Tobias,

is that what you want to do ?

(e.g. from the emcsim.C)

```
CbmRuntimeDb *rtdb=fRun->GetRuntimeDb();
CbmParRootFileIo* output=new CbmParRootFileIo(kParameterMerged);
output->open("simparams.root");
rtdb->setOutput(output);
rtdb->saveOutput();
rtdb->print();
```

And there is more in the wiki.

http://wiki.gsi.de/cgi-bin/view/Pandacomputing/PandaRootDatabase

(I just put there the upper part, the more detailed part was written by an unknown author...)

Soeren

Subject: Re: Runtime DB
Posted by Dima Melnychuk on Tue, 29 May 2007 17:00:04 GMT
View Forum Message <> Reply to Message

Hi Tobias,
The most detailed description concerning runtime DB is in the presentation of Denis Bertini which he gave at Frascati workshop.
You can find his talk on

http://wiki.gsi.de/cgi-bin/view/Pandacomputing/TutorialWorkshopFrascati2 006

under the title "Parameter Database Tutorial".

Dima

Subject: Re: Runtime DB
Posted by Sebastian Neubert on Fri, 26 Oct 2007 09:58:48 GMT
View Forum Message <> Reply to Message

Hi!

I am trying to use the file I/O of the RuntimeDB following the tutorial from the Wiki and looking at what was done for the emc and the mvd.

I note that in contrast to what the tutorial says both detectors do not use the activateParIo(..) method. I conclude that it is not really needed anymore (??)

However - although I do not find differences in my implementation to emc or mvd - the TpcDigiPar container is not being initialized and when I try to access it in the Init() of a task it crashes consequently.

Any ideas?

Sebastian.

Subject: Re: Runtime DB
Posted by Sebastian Neubert on Fri, 26 Oct 2007 11:43:23 GMT
View Forum Message <> Reply to Message

Hi!

I have tracked down the problem to the fact that i want to store strings in the parameter file (and this does not work). How can I do that?

Cheers! Sebastian.

Subject: Re: Runtime DB
Posted by Sebastian Neubert on Fri, 26 Oct 2007 11:55:15 GMT
View Forum Message <> Reply to Message

Hi!

For the moment I will create a workaround using integers as indexecs into (yet another) config file where I store the strings.

This is quite cumbersome so I request that strings be supported by the database if there is no major point against this.

Cheers! Sebastian.

Subject: Re: Runtime DB
Posted by Sebastian Neubert on Fri, 26 Oct 2007 16:02:08 GMT
View Forum Message <> Reply to Message

Yet another question:

I want to store my TpcDigiPar in the root db-file, so that it will be automatically available in the reconstruction. The storage works.

However it seems as if I cannot initialize digipar from the root file. My question: this should be no general problem, should it?

Cheers! Sebastian.

Subject: Re: Runtime DB
Posted by Sebastian Neubert on Tue, 30 Oct 2007 10:10:18 GMT
View Forum Message <> Reply to Message

Hi!

Storage and read from ROOT file solved.

S.

Subject: Re: Runtime DB
Posted by Sebastian Neubert on Tue, 30 Oct 2007 10:20:49 GMT
View Forum Message <> Reply to Message

A General question:

How does the RuntimeDB handle multiple reprocessing of the same(!) run?

The following use-case is very common at the moment: I have an MC-data file and want to digitize it several times with different parameter sets to do Detector-Studies. In the reconstruction and in the analysis stages I might also want to try out several sets of settings. Moreover I expect the database to ensure, that when I look at one event for analysis I will get the correct parameters, which have been used for digitization (reco... etc.).

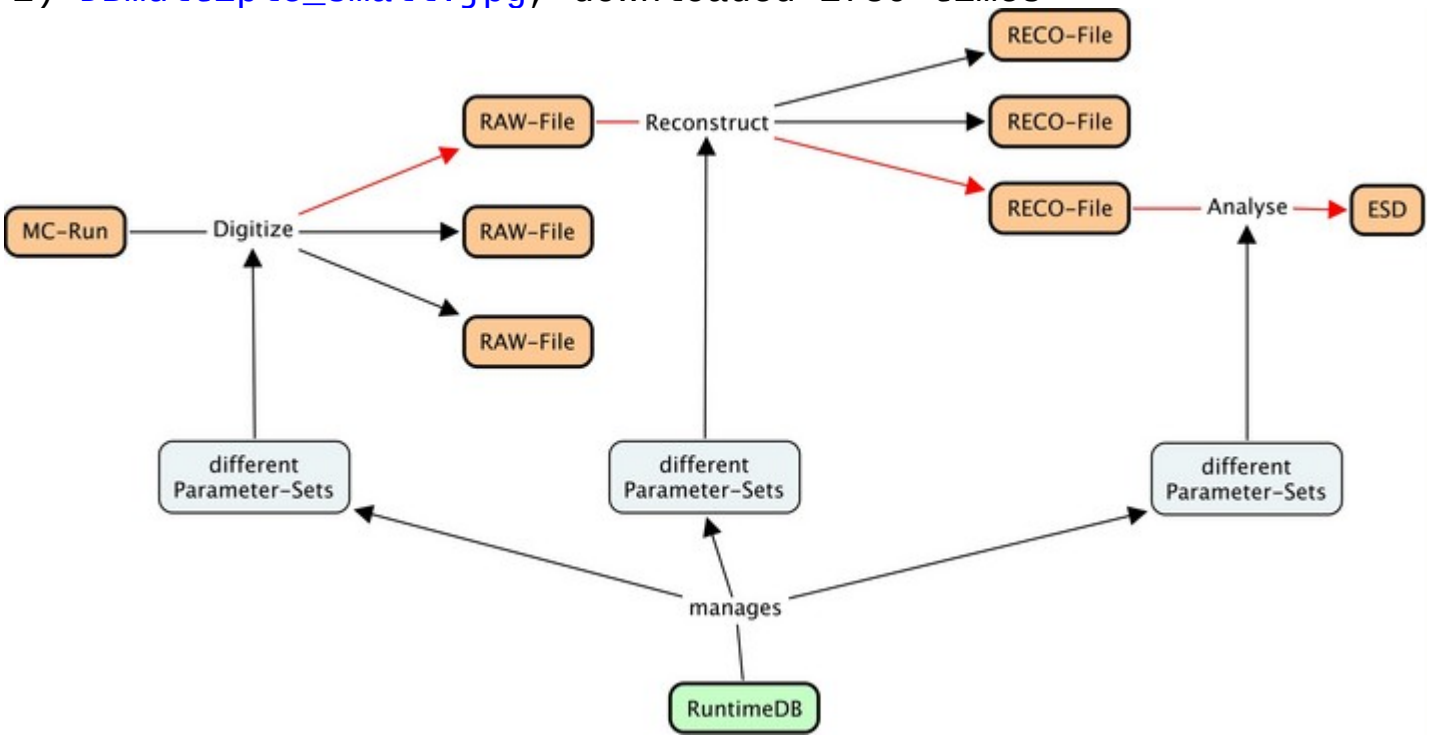
How is this handled?

Cheers! Sebastian.

Illustration of problem:

File Attachments

1) DBmultiple_small.jpg, downloaded 1780 times



Subject: Re: Runtime DB
Posted by Dima Melnychuk on Tue, 30 Oct 2007 11:04:31 GMT
View Forum Message <> Reply to Message

Hi Sebastian,

May be I do not undestand your question correctly, but I can guess that it is solved in the way that each set of parameters can have different versions.

And you can use spesific set of parameters by asking spesific version.
You can look into presentation of Denis Bertini from Frascaty Workshop, at the page 10 it is described version management with root files.

Or do you mean something different?

Dima

Subject: Re: Runtime DB
Posted by Sebastian Neubert on Tue, 30 Oct 2007 11:38:46 GMT
View Forum Message <> Reply to Message

Hi Dima!

You are absulotely correct, that there is support for different versions.

BUT: a version in the root file corresponds to one RunID. The RunID is assigned during the monte carlo. That means all files that are produced from one mc-file will have the same runID -> will use the same version of the parameters.

At least that is how I understand it!

Cheers! Sebastian.

Subject: Re: Runtime DB
Posted by Dima Melnychuk on Wed, 31 Oct 2007 13:05:48 GMT
View Forum Message <> Reply to Message

Hi Sebastian,

Did you manage to store strings in Runtime-DB already?
I tried and it works for me. If you haven't solved it yet I can shortly describe how I did it.

Dima

Subject: Re: Runtime DB
Posted by Sebastian Neubert on Wed, 31 Oct 2007 13:12:28 GMT
View Forum Message <> Reply to Message

Cool Dima!
Tell me, please!

Subject: Re: Runtime DB
Posted by Dima Melnychuk on Wed, 31 Oct 2007 14:43:15 GMT
View Forum Message <> Reply to Message

I will describe how I stored string in parameter file on example of emc. I don't uppload the code to svn, since it was just a test.

1. In PndEmcRecoPar.h defined data member

Text_t fEmcClusterPosMetodText[20];

2. In PndEmcRecoPar.cxx, in PndEmcRecoPar::putParams(CbmParamList *list)

list->add("EmcClusterPosMethodText",fEmcClusterPosMethodText,20);

3. In PndEmcRecoPar::getParams(CbmParamList *list)

if (!list->fill("EmcClusterPosMethodText",fEmcClusterPosMethodText,20)) return kFALSE;

4. In emc.par file defined the line
EmcClusterPosMethodText:s lilo

And that's it in principle.

Dima

Subject: Re: Runtime DB
Posted by Sebastian Neubert on Wed, 31 Oct 2007 14:50:44 GMT
View Forum Message <> Reply to Message

Hi Dima!

This is exaclty how I tried it... strange. I probably made a stupid mistake. Thanks anyway! I keep on trying!

Cheers! Sebastian.

Subject: Re: Runtime DB
Posted by Mohammad Al-Turany on Tue, 06 Nov 2007 14:55:39 GMT
View Forum Message <> Reply to Message

Hi,

Sorry that I did not answer this before, but I saw it first last
week and I was very busy with other problems, any way I have a
few remarks:

BUT: a version in the root file corresponds to one RunID. The
RunID is assigned during the monte carlo. That means all files
that are produced from one mc-file will have the same runID->
will use the same version of the parameters.

This is not completely true! when you write out your parameters you call
CbmParSet::setInputVersion(Int_t, Int_t):
e.g:

```
PndConstField *fc= (PndConstField *)fField;
PndConstPar *cp = (PndConstPar*)
rtdb->getContainer("PndConstPar");
cp->SetParameters(fc);
cp->setInputVersion(fRun->GetRunId(),1);
```

The second integer is the version number!

So for one runId you can have more versions, technically speaking, this is presented by a cycle number in ROOT file. the disadvantage of this is that each time you create a new file it will start counting from 1! and that you have to take care your self about this version number (i.e two root files could have same runId and version number eventhough different parameter sets).

The other and better way to do this is to use the so called context, where you can set a text description to your parameter set. This will be saved with the same RunId but different context which you can use in the initialization. I will try to write a simple example of how to use this as soon as possible!

regards

Mohammad

Subject: Re: Runtime DB
Posted by Sebastian Neubert on Tue, 06 Nov 2007 15:11:28 GMT
View Forum Message <> Reply to Message

Hi Mohammed!

Thanks for your reply!
I understand it much better now. So the runId is not mapped to the version number.

The possibility to use a certain context seems very interesting to me! I had seen it in the code, but was not sure how to make use of it. An example would be most welcome!

Cheers! Sebastian.

Subject: Re: Runtime DB
Posted by Sebastian Neubert on Sat, 10 Nov 2007 09:33:18 GMT
View Forum Message <> Reply to Message

Hey!

Just a follow up question: You said, that when using version ids the user has to keep track of the version himself.
Is the same true when using the context?

If so, we urgently need to discuss the database issue. I am running detector study jobs and already begin to feel the need for database support.

To be more specific: I would like the database to deliver the right parameters for an event without me having to worry about it. I would like to have this functionality:

* Have ONE database which is always used. The name of the database(file) is the only thing I specify in my scipts.

* To add parameters to the database I use the available mechanism to create Par-Objects from text-file.

* I expect those Par-Objects to be available in any proceeding step of data-treatment, without any more user intervention.

If any of this is not supported please tell me soon, because then I would need to develop something!

Cheers! Sebastian.
