Subject: Channel / Crystal / Bar / Detector numbers
Posted by Ralf Plag on Thu, 04 Feb 2016 10:29:08 GMT
View Forum Message <> Reply to Message

Hi All,

how do we store channel / crystal / bar or detector numbers: 1-based or 0-based?

In other words: Should my 4 LOS channels count from 1..4 (nice for the user in front of root) or from 0..3 (nice for the programmer)?

Is there a already a rule?

Cheers,
Ralf

Subject: Re: Channel / Crystal / Bar / Detector numbers
Posted by Dmytro Kresan on Fri, 05 Feb 2016 07:39:33 GMT
View Forum Message <> Reply to Message

Hi Ralf,

to my opinion, channel / module / detector indexing is something code internal, which does not appear on the plots with final results.
to my taste, since we are writing c++ code, and often use GetChannelId() directly as index in the array, I would stick to 0 based numbering.

How is it done in Ucesb? Because r3broot readers receive already mapped data.

Cheers,
Dima

Subject: Re: Channel / Crystal / Bar / Detector numbers
Posted by Ralf Plag on Fri, 05 Feb 2016 08:21:26 GMT
View Forum Message <> Reply to Message

Hi Dima,

right, they should not appear in final plots, but whenever you look at a cbmsim-Tree, they appear.
I have no preference, I only think we should stick to a standard to keep the confusion at a minimum level.

The output of ucesb depends on the mapping, so there we have the choice. As I have seen so far, channel numbers are 1-based there.
land02 was actually quite clever and automatically added +1 to channel numbers when writing into a TTree. So they were 0-based in the code and 1-based for the user in the TTree.

How is it done for Neuland, Califa, and others?

Cheers,
Ralf

Subject: Re: Channel / Crystal / Bar / Detector numbers
Posted by Hector Alvarez Pol on Fri, 05 Feb 2016 09:56:51 GMT
View Forum Message <> Reply to Message

Hi,

for CALIFA, crystals numbers in the TTree start in 1 (crystalId runs from 1 to 1952 in the barrel).
Other data (crystalType, crystalCopy, ...) also starts at 1 at the TTree level.

This was made just to match the crystal documentation/nomenclature from CAD and simplicity for the users.
If we would prefer a general scheme starting at 0, changing to a new standard is not a huge issue, ... we could fix and document.

Best regards,
Héctor

Subject: Re: Channel / Crystal / Bar / Detector numbers
Posted by Hector Alvarez Pol on Fri, 05 Feb 2016 10:00:04 GMT
View Forum Message <> Reply to Message

Hi,

I was not clear enough, I guess... despite it is not a big issue to change code for one or other option, my vote would be to number detector elements starting by 1 at TTree level.
I think it is more clear for users, once is properly documented. Developers should take care of internals of array to id conversion.

Best regards,
Héctor

Subject: Re: Channel / Crystal / Bar / Detector numbers
Posted by Dmytro Kresan on Fri, 05 Feb 2016 10:14:59 GMT
View Forum Message <> Reply to Message

Content of data members of a class, as it is at the runtime, is directly streamed to the output tree. The only possibility is private data member having 1-based values, and public Getter() function making shift to 0, to be used in the code. Making things already not transparent.

Summary: mapping is mostly 1-based, CAD also, currently CALIFA and NeuLAND in r3broot as well, and users prefer 1-based as well. This makes 5 points for 1-based agains only my

personal taste.

Let us fix and document "physical" indexing (starting at 1), and programmers have to take care when accessing arrays. I am not very much for implementing automatic conversion, as explained before.

Cheers,
Dima

Subject: Re: Channel / Crystal / Bar / Detector numbers
Posted by Ralf Plag on Fri, 05 Feb 2016 10:18:48 GMT
View Forum Message <> Reply to Message

I agree.
Ralf

Subject: Re: Channel / Crystal / Bar / Detector numbers
Posted by Dmytro Kresan on Fri, 05 Feb 2016 10:25:54 GMT
View Forum Message <> Reply to Message

We are slowly collecting enough material to write up "Computing rules for R3BRoot"
