Subject: Constant magnetic field instead Solenoid Field maps
Posted by Anastasia Karavdina on Mon, 01 Jul 2013 19:36:46 GMT
View Forum Message <> Reply to Message

Dear all,
As the luminosity detector group learnt during the last coll.meeting, solenoid magnetic field can't be 2T in all beam momentum range and for energies below ~3 GeV is should be 1T.

As far as I know, there is unique set of maps for solenoid field which is used for all energies. Is it possible to generate field maps for 1.5 GeV with 1T solenoid field? Whom we should ask to do it?

In meanwhile, since for LMD we have many troubles @1.5GeV, I'm trying to do tests with constant field instead loading maps for Solenoid Field. The idea is use Bz=1T for this constant field. But for cross-check with results obtained with standard solenoid maps I'm using 2T in 1st test.

The part of macro code looks like this:

```
 PndMultiField *fField= new PndMultiField();
 PndTransMap *map_t = new PndTransMap("TransMap", "R");
  PndDipoleMap *map_d1 = new PndDipoleMap("DipoleMap1", "R");
  PndDipoleMap *map_d2 = new PndDipoleMap("DipoleMap2", "R");
  fField->AddField(map_t);
  fField->AddField(map_d1);
  fField->AddField(map_d2);

  PndConstField *fSolField=new PndConstField();
  fSolField->SetField(0,0,20); // values are in kG //for cross-check with results from maps set Bz=2T
  fSolField->SetFieldRegion(-240,240,-240,240,-172,283.7);//z range is sum from Solenoid#1-#4 maps
  fField->AddField(fSolField);

  fRun->SetField(fField);
```

The simulation with such fields set is running, but I see many messages:

GetBz Should be implimented

Does it mean that I didn't something wrong with constant mag. field?

Best regards,
Anastasia.

Subject: Re: Constant magnetic field instead Solenoid Field maps
Posted by Ralf Kliemt on Tue, 02 Jul 2013 05:53:26 GMT

View Forum Message <> Reply to Message

Hello Anastasia,

The GetBz() function in FairField is not implemented. Please get your B Field like that (important are the double arrays):

```
double pnt[3]={0.,0.,0.}; //Position where to get field strength
double Bf[3]; //result goes here
// retrieve the field from the framework
FairRunAna::Instance()->GetField()->GetFieldValue ( pnt, Bf );  //[kGs]
return Bf[2]; //OK, Bf[2] is your Bz
```

Please mind the units.

In Analysis macros you can use RhoCalculationTools::GetBz(TVector3 pos).

Cheers.
Ralf

Subject: Re: Constant magnetic field instead Solenoid Field maps
Posted by Anastasia Karavdina on Tue, 02 Jul 2013 07:30:31 GMT
View Forum Message <> Reply to Message

Hi Ralf!
Thank you for your suggestion. The problem that I didn't call GetBz() by myself. It's called somewhere during generation with FairBoxGenerator. Since this errors message doesn't contain any information where it is produced, I don't know where to insert piece of code which you suggested.

Cheers,
Anastasia.

Subject: Re: Constant magnetic field instead Solenoid Field maps
Posted by StefanoSpataro on Tue, 02 Jul 2013 08:24:05 GMT
View Forum Message <> Reply to Message

Hi,
can you post your complete macro? The box generator does not the field call. The point is that in MultField the GetBxyz is not implemented, you should use another function to get such values. Who is asking for the field value?

About the field calculations, you should ask Jost Luehning at GSI. Not only the solenoid map should be calculated, but also the transient one.

Subject: Re: Constant magnetic field instead Solenoid Field maps
Posted by Anastasia Karavdina on Tue, 02 Jul 2013 08:33:51 GMT
View Forum Message <> Reply to Message

Hi Stefano!
Thanks for your reply! Please, find attached macro which I'm using for simulation.
I still don't know who is calling this GetBz method, but we are searching for it.

Cheers,
Anastasia.

File Attachments
1) runLumiPixel0SimBox2T.C, downloaded 462 times

Subject: Re: Constant magnetic field instead Solenoid Field maps
Posted by Anastasia Karavdina on Tue, 02 Jul 2013 08:58:34 GMT
View Forum Message <> Reply to Message

Ok, the message "GetBz Should be implimented" is misleading since it's printed from GetBxyz method, which is indeed isn't implemented in PndConstField. Also it isn't implemented in PndFieldMap class, but there is no messages with maps fields usage.
Seems like this call happens in

```
void PndMultiField::GetFieldValue(const Double_t point[3], Double_t* bField)
{

   PndRegion *fReg=0;
   FairField *fField=0;
   for (fMapIter=fFieldMaps.begin(); fMapIter!= fFieldMaps.end();fMapIter++ ){
      fReg=fMapIter->first;
      if(fReg->IsInside(point[2])){
         fField=fMapIter->second;
  break;
      }
   }
  if(fField){
     fField->GetBxyz(point, bField);
  }else{
    bField[0] = 0;
    bField[1] = 0;
    bField[2] = 0;
  }
}
```

But if I understand this code correctly is should call GetBxyz for both PndConstField as well as PndFieldMap. Why there is no error messages when PndFieldMap is used?

Subject: Re: Constant magnetic field instead Solenoid Field maps
Posted by Anastasia Karavdina on Tue, 02 Jul 2013 09:43:17 GMT
View Forum Message <> Reply to Message

Well, I was wrong about GetBxyz method implementation in case field maps classes, actually it is implemented there.

For PndConstField I propose following implementation of this method (which is basicly merge of GetBx(), GetBy(),GetBz()):

```
void PndConstField::GetBxyz(const Double_t point[3], Double_t* bField){
  if ( point [0] < fXmin  ||  point [0] > fXmax  ||
       point [1] < fYmin  ||  point [1] > fYmax  ||
       point [2] < fZmin  ||  point [2] > fZmax ) {
    bField[0]=0;     bField[1]=0;    bField[2]=0;
  }
  else{
    bField[0]=fBx;     bField[1]=fBy;    bField[2]=fBz;
  }
}
```

Can it be insert in svn version of code, please? Stefano, I can send changed source and header files if they needed.

Cheers,
Anastasia

Subject: Re: Constant magnetic field instead Solenoid Field maps
Posted by Mohammad Al-Turany on Wed, 03 Jul 2013 07:37:04 GMT
View Forum Message <> Reply to Message

Hallo,

let me try to explain you what is the problem here, before telling you that it is now completely solved in svn 20585!

The method GetFieldValue is called from VMC (simulation) so we have to keep the name and argument as they are. However some time one is interested in only one component of the field, that is why we implement the GetBx, GetBy and GetBz methods, now each of these methods check the boundary of the field map so when you need all of them it make no sense to check the same boundary three times, that is why we introduce the GetBxyz for the field maps but never for the constant field.

Now since Anastasia is mixing a constant field with maps this problem show up and exactly in the function PndMultiField::GetFieldValue as she find out, normally or till now we did not need to implement the GetBxyz for the constant field (we never have this use case), most of the people are using the scale of the field to change the whole map if needed, maybe this is also an option for you to test! you can simply use

void PndFieldMap::SetScale(Double_t factor)

This will scale the field map you use by this factor (all field components will be scaled for the

map)

In any case one should also implement the function for GetBxyz for the const field also to solve the problem in your use case. This is all done now and I also change the output to have more meaningful warnings!

best regards,

Mohammad

Subject: Re: Constant magnetic field instead Solenoid Field maps
Posted by Anastasia Karavdina on Wed, 03 Jul 2013 07:52:27 GMT
View Forum Message <> Reply to Message

Hi Mohammad,
Thanks a lot for quick reaction and changes in the code!
Also it's good to know about possibility to scale the maps. I'll certainly try it.

Cheers,
Anastasia.
