

## 猪头 Geant4讲座第五讲——材料定义

管理提醒：本帖被 zhutou 执行提前操作(2008-10-08)

本讲座为蒙卡学术论坛（52mc.net）专题讲座，任何人未经作者本人许可不得转载至其他论坛，作者保留追究转载者相关责任的权利！

此帖售价 20 金币,已有 44 人购买

几何结构类（**DetectorConstruction**）属于强制初始化类，其主要功能是构建模拟问题的几何结构，包括各部分的材料、形状、尺寸、位置等信息。

因此，在这个类里面我们就必须完成上述几个信息的设置工作。

首先，材料定义。

材料可以分为单质和化合物（混合物）两种。而不管是单质还是化合物都是由元素组成的，因此在定义材料前，必须首先定义元素。而元素的定义将决定在模拟过程中需要使用的截面库的选择（大部分是自动选择的，这里不重点讲）。

那么下面我们来看如何定义元素。

我们知道，每一种元素都可能有多同位素，但是所有这些同位素的原子序数（核内质子数）都是相同的，其摩尔质量也可以根据各个同位素所占份额计算出来。因此，只需要有原子序数  $Z$  和摩尔质量  $A$  就可以定义出一个元素。这就是元素的直接定义法，参考 `$G4INSTLL/source/materials/include/G4Element.hh`，如下：

```
// Constructor to Build an element directly; no reference to isotopes
//
G4Element(const G4String& name, //its name
           const G4String& symbol, //its symbol
           G4double Zeff, //atomic number
           G4double Aeff); //mass of mole
```

其中元素名称和符号只是个标记，并不会影响元素的物理性质。

`$G4INSTALL/example/novice/N02`中氮元素的定义就是采用的直接定义法。

```
G4Element* N = new G4Element("Nitrogen", "N", z=7., a= 14.01*g/mole);
```

此外，既然每种元素都是由不同的同位素组成的，那如果事先定义了同位素，加上每个同位素所占份额不也可以确定一种元素，而不必麻烦地去计算摩尔质量吗？

确实如此，在 **Geant4**中同样提供了另一种定义元素的方法，我将之称为间接定义法。

同样参考`$G4INSTLL/source/materials/include/G4Element.hh`，如下：

```
// Constructor to Build an element from isotopes via AddIsotope
//
G4Element(const G4String& name, //its name
           const G4String& symbol, //its symbol
           G4int nblsotopes); //nb of isotopes
void AddIsotope(G4Isotope* isotope, //isotope
                G4double RelativeAbundance); //fraction of nb of
```

//atomes

per volume

而同位素的定义则参考\$G4INSTLL/source/materials/include/G4Isotope.hh

```
G4Isotope(const G4String& name, //its name
           G4int      z, //atomic number
           G4int      n, //number of nucleons
           G4double   a = 0.); //mass of mole
```

\$G4INSTALL/example/novice/N03中铀元素的定义就是采用的间接定义法。

```
// define an Element from isotopes, by relative abundance
```

```
//
```

```
G4Isotope* U5 = new G4Isotope("U235", iz=92, n=235, a=235.01*g/mole);
G4Isotope* U8 = new G4Isotope("U238", iz=92, n=238, a=238.03*g/mole);
G4Element* U  = new G4Element("enriched Uranium",symbol="U",ncomponents=2);
U->AddIsotope(U5, abundance= 90.*perCent);
U->AddIsotope(U8, abundance= 10.*perCent);
```

定义完元素之后我们就可以定义材料了。

前面我们说了，材料可以分为单质（只有一种元素）和化合物（或混合物，含有两种或两种以上的元素）。

不管是单质还是化合物，如果我们知道其中每种元素的份额（单质可以认为其组成元素的份额是100%），那我们就可以确定这种材料的组成了。再加上密度等信息就可以确定这种材料的具体状态了。这就是 **Geant4**中的一种材料定义法。而 **Geant4**中为了方便定义材料，将份额还分为了两种，分别是原子数份额和质量份额。

定义方法参考\$G4INSTLL/source/materials/include/G4Material.hh 如下：

```
// Constructor to create a material from a combination of elements
// and/or materials subsequently added via AddElement and/or AddMaterial
//
G4Material(const G4String& name, //its name
           G4double   density, //density
           G4int      nComponents, //nbOfComponents
           G4State    state = kStateUndefined, //solid,gas
           G4double   temp = STP_Temperature, //temperature
           G4double   pressure = STP_Pressure); //pressure

//
// Add an element, giving number of atoms
//
void AddElement(G4Element* element, //the element
               G4int      nAtoms); //nb of atoms in
// a molecule

//
// Add an element or material, giving fraction of mass
//
void AddElement (G4Element* element , //the element
                G4double   fraction); //fractionOfMass
```

\$G4INSTALL/example/novice/N03中二氧化硅就是采用的原子数份额定义的，而空气则是采用的质量份额定义的。

```
G4Material* SiO2 =
```

```
new G4Material("quartz",density= 2.200*g/cm3, ncomponents=2);
```

```
SiO2->AddElement(Si, natoms=1);
```

```
SiO2->AddElement(O , natoms=2);
```

```
G4Material* Air =
```

```
new G4Material("Air" , density= 1.290*mg/cm3, ncomponents=2);
```

```
Air->AddElement(N, fractionmass=0.7);
```

```
Air->AddElement(O, fractionmass=0.3);
```

值得一提的是，在 Geant4中为了定义复杂的混合物，还提供了

```
AddMaterial(G4Material* material, //the material
```

```
G4double fraction); //fractionOfMass
```

函数，通过这个函数可以将已经定义好的材料作为新材料的一种组成直接添加。

如\$G4INSTALL/example/novice/N03中气凝胶就是由62.5%的二氧化硅、37.4%的水和0.1%的碳组成的，由于二氧化硅和水在之前都已经定义了，因此直接使用AddMaterial(G4Material\* material, G4double fraction)添加即可，而碳则是一种元素需要用 AddElement(G4Element\* element, G4double fraction)添加，如下：

```
G4Material* Aerog =
```

```
new G4Material("Aerogel", density= 0.200*g/cm3, ncomponents=3);
```

```
Aerog->AddMaterial(SiO2, fractionmass=62.5*perCent);
```

```
Aerog->AddMaterial(H2O , fractionmass=37.4*perCent);
```

```
Aerog->AddElement (C , fractionmass= 0.1*perCent);
```

而对于单质，由于是由一种元素组成的，因此可以将元素定义和材料定义合并，

参考\$G4INSTLL/source/materials/include/G4Material.hh 如下

```
// Constructor to create a material from scratch.
```

```
//
```

```
G4Material(const G4String& name, //its name
```

```
G4double z, //atomic number
```

```
G4double a, //mass of mole
```

```
G4double density, //density
```

```
G4State state = kStateUndefined, //solid,gas
```

```
G4double temp = STP_Temperature, //temperature
```

```
G4double pressure = STP_Pressure); //pressure
```

如\$G4INSTALL/example/novice/N03中铝材料就是直接定义的。

```
new G4Material("Aluminium", z=13., a=26.98*g/mole, density=2.700*g/cm3);
```

需要注意的是在采用直接定义法定义元素或者直接定义单质的时候，其中的原子序数的类型是 **G4double**，而不是 **G4int**，但是从\$G4INSTLL/source/materials/include/G4Element.cc 中看，这个变量最终似乎是被转换为 **G4int** 型处理的。由于时间关系，本人没有进行深入地

研究，有兴趣的同仁可以研究一下这里为什么不用 **G4int** 而用 **G4double**，是否还有别的含义。