

Materials

Geant4 Tutorial at Sao Paulo

5 February 2019

Dennis Wright



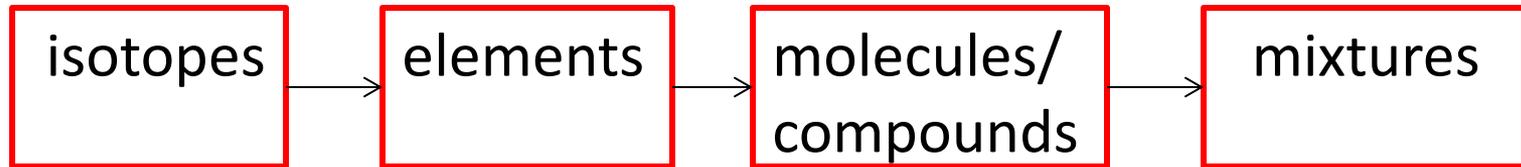
NATIONAL
ACCELERATOR
LABORATORY

Outline

- Materials in Geant4
- Material definition
- NIST material database

Materials in Geant4

- Geant4 materials (like those in the real world)
 - are made up of isotopes, elements, compounds (or molecules), mixtures of elements and/or compounds



- can be solid, liquid or gas (sorry, no plasma)
- may exist under various pressures, temperatures and densities

Materials in Geant4

- Geant4 allows the custom definition of materials
 - starting with elements: use `G4Element` class
 - compounds or molecules can be built by assigning two or more instances of `G4Element` to an object of the `G4Material` class
 - mixtures can be built by assigning two or more compounds or elements to an instance of `G4Material`
 - a `G4Material` can also be built from a single `G4Element`
 - optionally, you may define your own `G4Element` by assigning to it one or more instances of the class `G4Isotope`

Materials in Geant4

- Geant4 requires you to set at least one material condition
 - density
- The rest are optional
 - state (default is solid or gas, depending on density)
 - temperature (default = STP temperature = 273.15 K)
 - pressure (default = STP pressure = 100 kPascal = 1 atm)
- Along with normal stuff, you can define some strange things
 - gases far from STP
 - high pressure solids
 - low density liquids
- And there's a shortcut => NIST material database

Definition of Materials

- Let's start with a single-element material:

```
G4double density = 4.506*g/cm3;
```

```
G4double a = 47.867*g/mole;
```

```
G4Material* ti = new G4Material("pureTitanium", z=22, a, density);
```

- Vacuum is also useful

```
G4NISTManager* manager = G4NISTManager::GetPointer;
```

```
G4Material* vacuum = manager->FindOrBuildMaterial("G4_Galactic");
```

—for vacuum, use low density gas rather than density = 0

—“average” materials (e.g. $z = 25.7$) are not allowed

Definition of Materials: molecules

- A molecule is made of several elements, with the composition specified by the number of atoms

```
G4double a = 1.01*g/mole;
```

```
G4Element* elH = new G4Element("Hydrogen", "H", z=1, a);
```

```
a = 16.00*g/mole;
```

```
G4Element* elO = new G4Element("Oxygen", "O", z=8, a);
```

```
G4double density = 1.0*g/cm3;
```

```
G4int ncomp = 2;
```

```
G4Material* H2O = new G4Material("Water", density, ncomp);
```

```
G4int nAtoms;
```

```
H2O->AddElement(elH, nAtoms=2);
```

```
H2O->AddElement(elO, nAtoms=1);
```

Definition of Materials: mixtures (alloys)

- A mixture is similar to a molecule, except that materials and elements are combined instead of just elements

```
G4Element* elC = ... ;    // define carbon
```

```
G4Material* H2O = ... ;   // define molecule (previous page)
```

```
G4Material* SiO2 = ... ;  // define another molecule
```

```
G4double density = 0.20*g/cm3;
```

```
G4int ncomp = 3;
```

```
G4double fracMass;
```

```
G4Material* Aerog = new G4Material("Aerogel", density, ncomp);
```

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

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

```
Aerog->AddElement(elC, fracMass = 0.1*perCent);
```

Elements and Isotopes

- If you define an element, it is treated by default as if it has the natural isotope abundance
 - even if the g/mole value you enter is quite different from the natural abundance
 - hadronic code knows only how to deal with specific nuclides, not elements
- You can define an element with non-natural abundance by assigning to G4Element a list of G4Isotope instances
- Example: making nuclear fuel – start with isotopes

```
G4int z; G4int a;
```

```
G4Isotope* u235 = new G4Isotope("U235", z=92, a=235.,  
    235.044*g/mole);
```

```
G4Isotope* u238 = new G4Isotope("U238", z=92, a=238.,  
    238.051*g/mole);
```

Elements and Isotopes (continued)

- Example: use the isotopes to define enriched uranium for power generation:

```
G4int ncomp;
```

```
G4double abundance;
```

```
G4Element* enrichedU = new G4Element("enrichedU", ncomp=2);
```

```
enrichedU->AddIsotope(u235, abundance=5.0*perCent);
```

```
enrichedU->AddIsotope(u238, abundance=95.0*perCent);
```

- Make some fluorine, too

```
G4Element* eF = new G4Element("Fluorine", "F", 9., 18.998*g/mole);
```

Elements and Isotopes (continued)

- Example: now use F and enriched U to make nuclear fuel (UF_6)
G4double density;
G4Material* fuel = new G4Material("NuclearFuel", density = 5.09*g/cm3,
ncomp=2, kStateGas, 640*kelvin, 1.5e7*pascal) ;
fuel->AddElement(eIF, 6);
fuel->AddElement(enrichedU, 1);
- Note the red entries: state, temperature, pressure
 - these are optional parameters which default to STP if not specified

Using the NIST Material Database

- Most of the materials you will want to define are already done for you
 - also all elements with natural isotopic abundance
 - more than 3000 isotopes defined
- Geant4 has included these pre-defined materials from the NIST database
 - physics.nist.gov/PhysRefData
 - provides the best accuracy for major parameters
 - density
 - isotopic composition of elements
 - elemental composition of materials
 - mean ionization potential
 - chemical bonds

Pre-defined NIST Materials

- NIST elementary materials
 - up to $Z = 98$ (Cf)
- NIST compounds and mixtures
 - tissue equivalent plastic, dry air at sea level, many others
- HEP and nuclear materials
 - liquid Ar, PbWO_4 , CR39, etc.
- Space materials
 - Kevlar, Dacron and so on
- Biochemical materials
 - adipose tissue, cytosine, thymine, etc.
- 315 materials so far

How to Use Pre-defined Materials

- User interfaces to C++ code:

```
G4NISTManager* man = G4NistManager::Instance();
```

```
G4Element* elm = man->FindOrBuildElement("Chem symbol");
```

```
G4Element* elm = man->FindOrBuildElement(G4int Z);
```

```
G4Material* mat = man->FindOrBuildMaterial("Name");
```

```
G4Material* mat = man->ConstructNewMaterial("Name",  
                                           const std::vector<G4int>& Z,  
                                           const std::vector<G4double>& weight,  
                                           G4double density);
```

```
G4double isotopeMass = man->GetMass(G4int Z, G4int N);
```

- Elements built with natural isotope abundance

How to Use Pre-defined Materials

- User interfaces to Geant4 command line
 - list all NIST-defined elements:
`material/nist/printElement`
 - list all NIST-defined materials:
`material/nist/listMaterials`

Summary

- Geant4 allows you to define materials in terms of
 - isotopes (G4Isotope)
 - elements (G4Element)
 - materials (G4Material)
- And to set material conditions
 - density
 - state
 - temperature
 - pressure
- Whenever you can, use the pre-defined NIST database elements and materials
 - accurate and standardized