User Actions

Jonathan R. Madsen

Department of Nuclear Engineering
Texas A&M University
College Station, TX, USA 77843

madsen_jr@tamu.edu

Outline

Materials in Geant4

- Geant4 materials (like those in the real world) are made up of isotopes elements, compounds/molecules, mixtures of elements and/or compounds
- Materials states:
  - Solid
  - Liquid
  - Gas
  - No plasma
- Material properties:
  - Pressure
  - Temperature
  - Density
- Documentation reference: Book for Application Developers — Section 2.2

Building Custom Materials

- Custom Definition:
  - G4Isotope → construct G4Element
  - G4Element → construct G4Material
  - Compounds/molecules → assign two or more instances of G4Element to object of G4Material class.
    - Define by G4Element mass fraction
    - Define by G4Element number of atoms
  - G4Element can be built from single G4Isotope
  - G4Material can be built from single G4Element
  - Material property specification:
    - density (required)
    - state (optional) [default = solid/gas depending on density]
    - temperature (optional) [default = STP temperature = 273.15 K]
    - pressure (optional) [default = STP pressure = 100 kPa = 1 atm]

Building 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);
```

Building Materials - Mixtures

- 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
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 the g/mole value you enter is quite different from natural abundance
  - Hadronic code only knows 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 a nuclear fuel — start with isotopes

```
G4int z, a, ncomp;
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));
G4Element* enrichedU = new G4Element("EnrichedU", ncomp=2);
enrichedU->AddIsotope(u235, 10.0*perCent);
enrichedU->AddIsotope(u238, 90.0*perCent);
```

Elements and Isotopes (cont.)

- Build fluorine and make fuel (UF_6)

```
G4Element* elF = new G4Element("Fluorine", "F", 9., 18.988*(g/mole));
G4double density;
G4Material* fuel = new G4Material("Fuel", density=5.09*(g/cm3),
                                  ncomp=2, kStateGas, 640*kelvin,
                                  1.5e7*pascal);
fuel->AddElement(elF, 6);
fuel->AddElement(enrichedU, 1);
```

- State parameters (kStateSolid, ...) are optional parameters

Building Materials - NIST database

```
G4NistManager* nist = G4NistManager::Instance();
G4Material* vacuum = nist->FindOrBuildMaterial(``G4_Galactic'');
```

- G4NistManager should be accessed via static instance function
- For vacuum, use low density gas rather than zero density
- "average" materials (*e.g.* Z = 25.7) are not allowed

Using NIST database

- Most of the materials you want to define are already done for you
  - Also all elements with natural abundance
  - More that 3000 isotopes defined
- Geant4 has included the 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 pf materials
    - mean ionization potential
    - chemical bonds

Pre-defined NIST materials

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

Using NIST database

```
G4Element* el = nist->FindOrBuildElement("chemical_symbol",
                                         G4bool iso);
G4Material* mat = nist->FindOrBuildMaterial("name", G4bool iso);
G4Material* mat2 = nist->ConstructNewMaterial("name",
                            const std::vector<G4int>& Z,
                            const std::vector<G4double>& weight,
                            G4double density, G4bool iso);
```

- G4bool iso = true — element is built of isotopes with natural abundance
- G4bool iso = false — isotopes are not explicitly built

Exploring NIST database

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

Summary

- Geant4 allows you to define materials in terms of:
  - isotopes (G4Isotope)
  - elements (G4Element)
  - materials (G4Material)
- Set material conditions:
  - density
  - state
  - temperature
  - pressure
- Wherever possible, use the pre-defined NIST database for elements and materials
  - accurate and standardized