

Geant 4

Detector Description: Materials

<http://cern.ch/geant4>

PART I

Materials

- *The System of units & constants*
- *Definition of elements*
- *Materials and mixtures*
- *Some examples ...*
- *The NIST Data Base*

Units

- Geant4 has no default unit. To give a number, unit must be “multiplied” to the number.

- for example :

```
G4double width = 12.5*m;
```

```
G4double density = 2.7*g/cm3;
```

- If no unit is specified, the *internal* G4 unit will be used, but this is discouraged !
 - Almost all commonly used units are available.
 - The user can define new units.
 - Refer to CLHEP: `SystemOfUnits.h`
- Divide a variable by a unit you want to get.

```
G4cout << dE / MeV << “ (MeV)” << G4endl;
```

System of Units

- System of units are defined in CLHEP, based on:
 - millimetre (**mm**), nanosecond (**ns**), Mega eV (**MeV**), positron charge (**eplus**) degree Kelvin (**kelvin**), the amount of substance (**mole**), luminous intensity (**candela**), radian (**radian**), steradian (**steradian**)
- All other units are computed from the basic ones
- In output, Geant4 can choose the most appropriate unit to use. Just specify the *category* for the data (**Length**, **Time**, **Energy**, etc...):

```
G4cout << G4BestUnit (StepSize, "Length");
```

StepSize will be printed in **km**, **m**, **mm** or ... **fermi**, depending on its value

Defining new units

- New units can be defined directly as constants, or (suggested way) via `G4UnitDefinition`
 - `G4UnitDefinition` (name, symbol, category, value)
- Example (mass thickness):
 - `G4UnitDefinition` ("grampercm2", "g/cm2",
"MassThickness", g/cm2);
 - The new category "**MassThickness**" will be registered in the kernel in `G4UnitsTable`
- To print the list of units:
 - From the code
`G4UnitDefinition::PrintUnitsTable()`;
 - At run-time, as UI command:
`Idle> /units/list`

Definition of Materials

- Different kinds of materials can be defined:
 - isotopes <> **G4Isotope**
 - elements <> **G4Element**
 - molecules <> **G4Material**
 - compounds and mixtures <> **G4Material**
- Attributes associated:
 - temperature, pressure, state, density

Isotopes, Elements and Materials

- **G4Isotope** and **G4Element** describe the properties of the *atoms*:
 - Atomic number, number of nucleons, mass of a mole, shell energies
 - Cross-sections per atoms, etc...
- **G4Material** describes the *macroscopic* properties of the matter:
 - temperature, pressure, state, density
 - Radiation length, absorption length, etc...
- **G4Material** is the class used for geometry definition

Elements & Isotopes

- Isotopes can be assembled into elements

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

- ... building elements as follows:

```
G4Element (const G4String& name,  
          const G4String& symbol, // element symbol  
          G4int      nIso ); // # of isotopes  
G4Element::AddIsotope(G4Isotope* iso, // isotope  
                     G4double relAbund); // fraction of atoms  
                                         // per volume
```


Material of one element

- Single element material

```
G4double density = 1.390*g/cm3;
```

```
G4double a = 39.95*g/mole;
```

```
G4Material* lAr =
```

```
    new G4Material("liquidArgon", z=18., a, density);
```

- Prefer low-density material to vacuum

Material: molecule

- A Molecule is made of several elements (composition by number of atoms):

```
a = 1.01*g/mole;
G4Element* e1H =
    new G4Element ("Hydrogen", symbol="H", z=1., a);
a = 16.00*g/mole;
G4Element* e1O =
    new G4Element ("Oxygen", symbol="O", z=8., a);
density = 1.000*g/cm3;
G4Material* H2O =
    new G4Material ("Water", density, ncomp=2);
H2O->AddElement (e1H, natoms=2);
H2O->AddElement (e1O, natoms=1);
```

Material: compound

- Compound: composition by fraction of mass

```
a = 14.01*g/mole;
G4Element* e1N =
    new G4Element (name="Nitrogen", symbol="N", z= 7., a);
a = 16.00*g/mole;
G4Element* e1O =
    new G4Element (name="Oxygen", symbol="O", z= 8., a);
density = 1.290*mg/cm3;
G4Material* Air =
    new G4Material (name="Air", density, ncomponents=2);
Air->AddElement (e1N, 70.0*perCent);
Air->AddElement (e1O, 30.0*perCent);
```

Material: mixture

- Composition of compound materials

```
G4Element* e1C = ...; // define "carbon" element
G4Material* SiO2 = ...; // define "quartz" material
G4Material* H2O = ...; // define "water" material

density = 0.200*g/cm3;
G4Material* Aerog =
    new G4Material("Aerogel", density, ncomponents=3);
Aerog->AddMaterial(SiO2, fractionmass=62.5*perCent);
Aerog->AddMaterial(H2O , fractionmass=37.4*perCent);
Aerog->AddElement (e1C , fractionmass= 0.1*perCent);
```

Example: gas

- It may be necessary to specify temperature and pressure
 - (dE/dx computation affected)

```
G4double density = 27.*mg/cm3;  
G4double temperature = 325.*kelvin;  
G4double pressure = 50.*atmosphere;
```

```
G4Material* CO2 =  
    new G4Material("CarbonicGas", density, ncomponents=2  
                  kStateGas, temperature, pressure);  
CO2->AddElement(C, natoms = 1);  
CO2->AddElement(O, natoms = 2);
```

Example: vacuum

- Absolute vacuum does not exist. It is a gas at very low density !
 - Cannot define materials composed of multiple elements through **Z** or **A**, or with $\rho=0$

```
G4double atomicNumber = 1.;
G4double massOfMole = 1.008*g/mole;
G4double density = 1.e-25*g/cm3;
G4double temperature = 2.73*kelvin;
G4double pressure = 3.e-18*pascal;
G4Material* Vacuum =
    new G4Material("interGalactic", atomicNumber,
                  massOfMole, density, kStateGas,
                  temperature, pressure);
```

Gas mixture by volume ?

- Example: Ar (80%) CO₂ (20%)
 - In 10 moles of mixture :
8 moles of **Ar**, 2 moles of **CO₂**
 - The mixture is **equivalent** (for mass fraction) to the molecule **Ar₈ C₂ O₄**

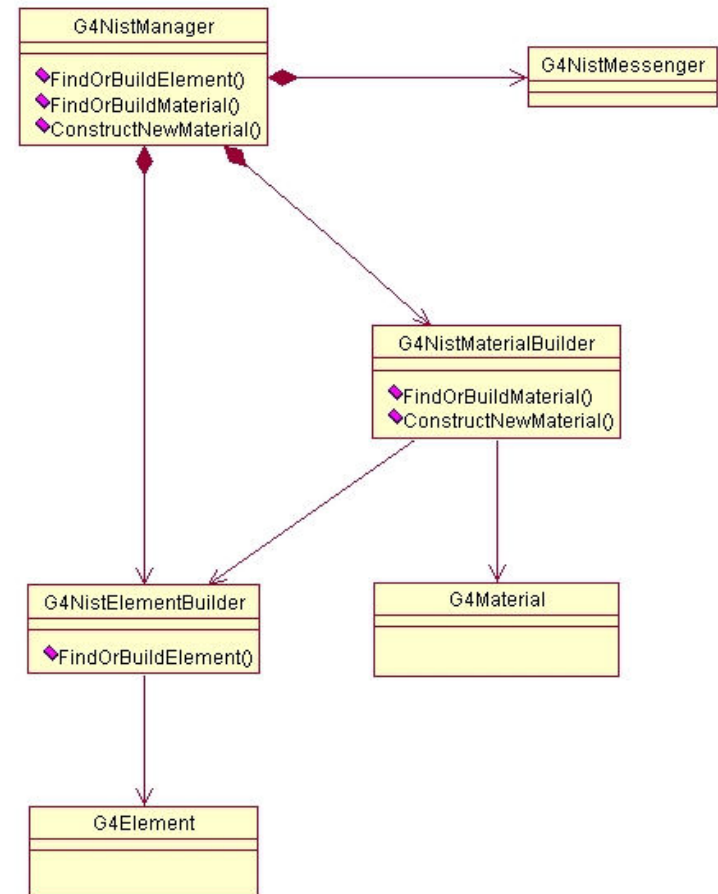
PART II

NIST Material Data-Base in Geant4

- *Class structure*
- *NIST Isotopes, Elements and Materials*
- *How to use it ...*

NIST Manager & Messenger

- NIST database for materials is imported inside Geant4
<http://physics.nist.gov/PhysRefData>
- Additional interfaces defined
- UI commands specific for handling materials
- The best accuracy for the most relevant parameters guaranteed:
 - Density
 - Mean excitation potential
 - Chemical bounds
 - Element composition
 - Isotope composition
 - Various corrections



NIST Elements & Isotopes

Z	A	m	error (%)	A_{eff}
14	Si 22	22.03453	(22)	28.0855(3)
	23	23.02552	(21)	
	24	24.011546	(21)	
	25	25.004107	(11)	
	26	25.992330	(3)	
	27	26.98670476	(17)	
	28	27.9769265327	(20)	92.2297 (7)
	29	28.97649472	(3)	4.6832 (5)
	30	29.97377022	(5)	3.0872 (5)
	31	30.97536327	(7)	
	32	31.9741481	(23)	
	33	32.978001	(17)	
	34	33.978576	(15)	
	35	34.984580	(40)	
	36	35.98669	(11)	
	37	36.99300	(13)	
	38	37.99598	(29)	
	39	39.00230	(43)	
	40	40.00580	(54)	
	41	41.01270	(64)	
	42	42.01610	(75)	

- Natural isotope compositions
- More than 3000 isotope masses
 - Used for elements definition

NIST Materials

=====
Elementary Materials from the NIST Data Base
=====

Z	Name	ChFormula	density(g/cm^3)	I(eV)
1	G4_H	H_2	8.3748e-05	19.2
2	G4_He		0.000166322	41.8
3	G4_Li		0.534	40
4	G4_Be		1.848	63.7
5	G4_B		2.37	76
6	G4_C		2	81
7	G4_N	N_2	0.0011652	82
8	G4_O	O_2	0.00133151	95
9	G4_F		0.00158029	115
10	G4_Ne		0.000838505	137
11	G4_Na		0.971	149
12	G4_Mg		1.74	156
13	G4_Al		2.6989	166
14	G4_Si		2.33	173

- NIST Elementary materials:
 - H → Cf (Z = 1 → 98)
- NIST compounds:
 - e.g. "G4_ADIPOSE_TISSUE_IRCP"
- HEP and Nuclear materials:
 - e.g. Liquid Ar, PbWO
- It is possible to build mixtures of NIST and user-defined materials

=====
Compound Materials from the NIST Data Base
=====

N	Name	ChFormula	density(g/cm^3)	I(eV)
13	G4_Adipose_Tissue		0.92	63.2
1		0.119477		
6		0.63724		
7		0.00797		
8		0.232333		
11		0.0005		
12		2e-05		
15		0.00016		
16		0.00073		
17		0.00119		
19		0.00032		
20		2e-05		
26		2e-05		
30		2e-05		
4	G4_Air		0.00120479	85.7
6		0.000124		
7		0.755268		
8		0.231781		
18		0.012827		
2	G4_CsI		4.51	553.1
53		0.47692		
55		0.52308		

How to use the NIST DB

- No need to predefine elements and materials
- Retrieve materials from NIST manager:

```
G4NistManager* manager = G4NistManager::Instance();
```

```
G4Material* H2O = manager->FindOrBuildMaterial("G4_WATER");
```

```
G4Material* mat = manager->ConstructNewMaterial("name",  
                                             const std::vector<G4String>& elements,  
                                             const std::vector<G4double>& weights,  
                                             G4double density, G4bool isotopes);
```

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

- Some UI commands ...

```
/material/nist/printElement ← print defined elements
```

```
/material/nist/listMaterials ← print defined materials
```

Material Scanner

- Tool to measure material thickness in units of geometrical length, radiation length and interaction length
 - Region sensitive: you can measure the thickness of one particular region
- `/control/matScan`
 - **scan** - *Start material scanning*
 - **theta** - *Define theta range*
 - **phi** - *Define phi range*
 - **singleMeasure** - *Measure thickness for one particular direction*
 - **eyePosition** - *Define the eye position*
 - **regionSensitive** - *Set region sensitivity*
 - **region** - *Define region name to be scanned*

Exercise 1a

- *Materials*