Experimental Techniques in Particle Physics

# Geant4: Materials

Andreas Nowack
nowack@physik.rwth-aachen.de
RWTH Aachen University

**WS 2020/21**

Quick Intro to
Geant 4

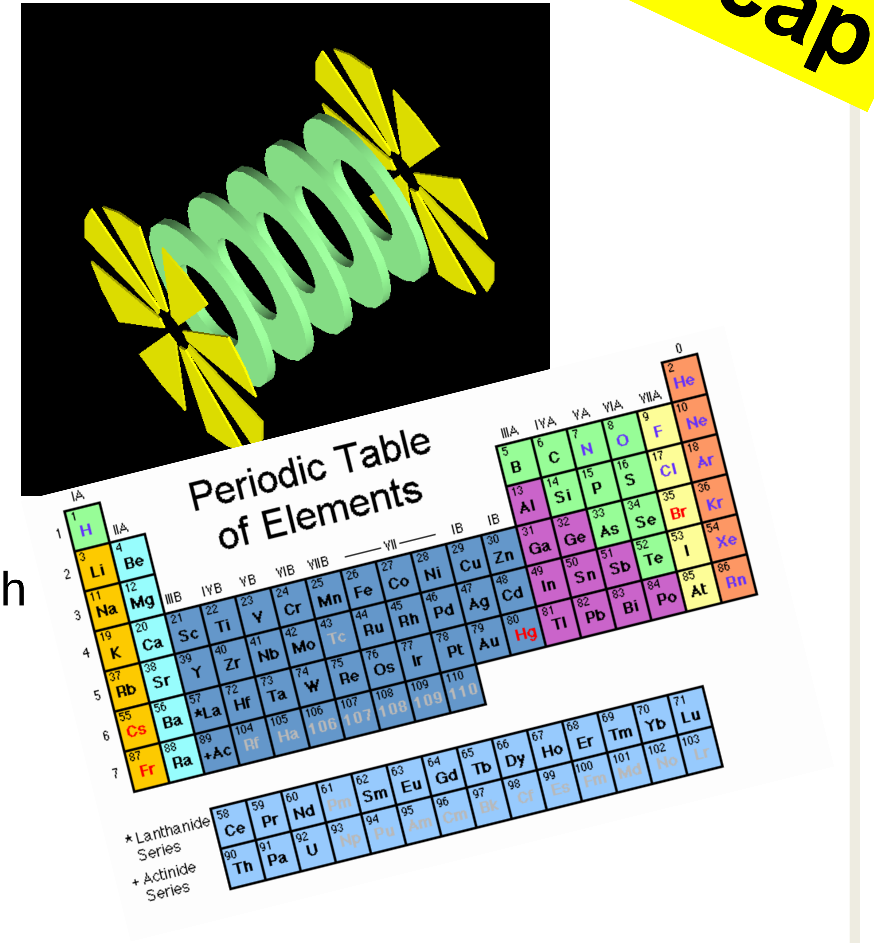
# Geometry

Recap

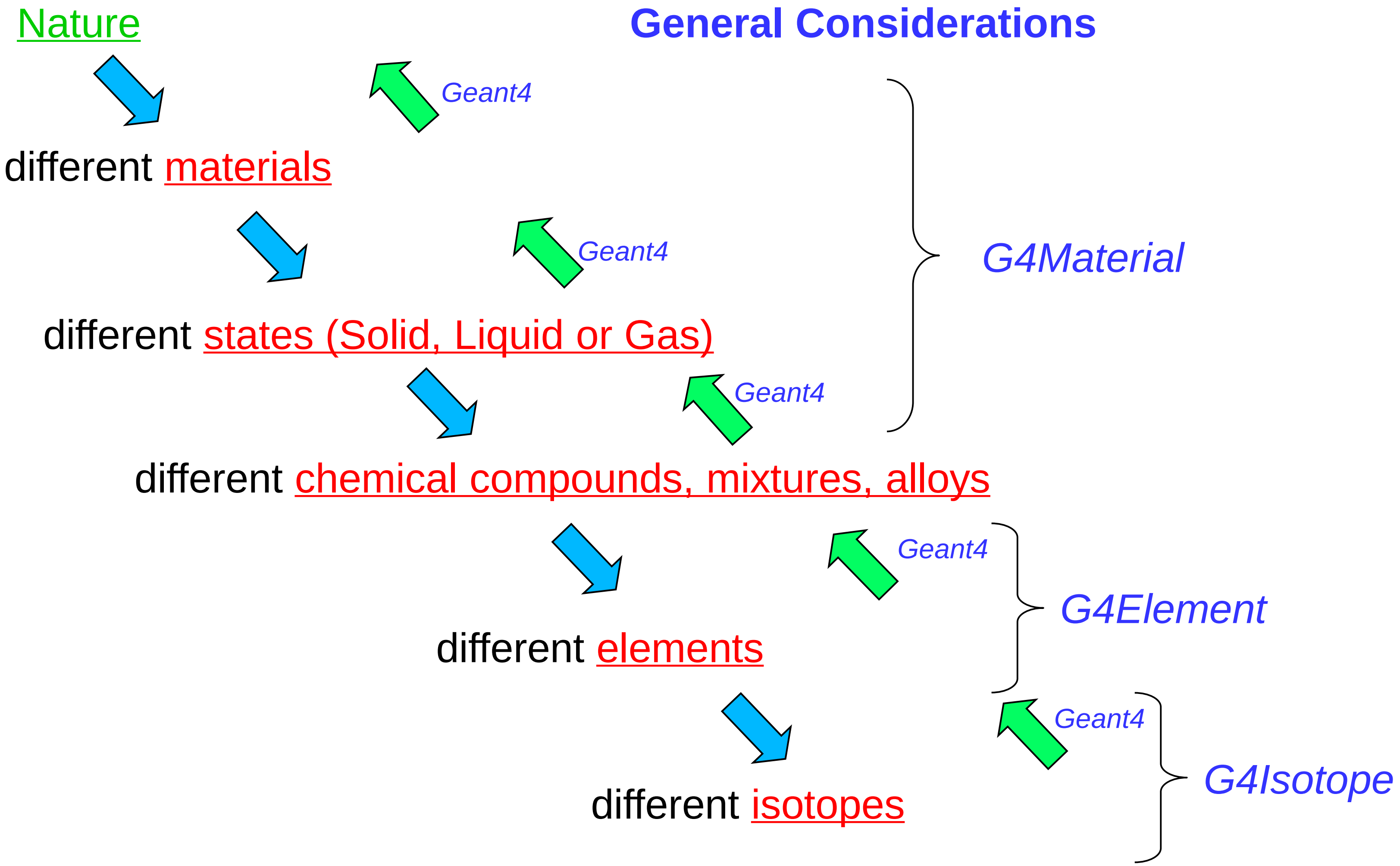
- So far:
  - simple geometrical shapes
  - Boolean operations
  - logical volumes
  - physical volumes
  - hierarchy of volumes
  - replicas

- We are able to construct complex geometrical set-ups.
- But these set-ups do not interact with particles.

- Physical properties have to be specified.



# Materials, Elements, and Isotopes



Nature

General Considerations

Geant4

different materials

Geant4

different states (Solid, Liquid or Gas)

Geant4

different chemical compounds, mixtures, alloys

G4Material

Geant4

different elements

G4Element

Geant4

different isotopes

G4Isotope

# Materials, Elements, and Isotopes

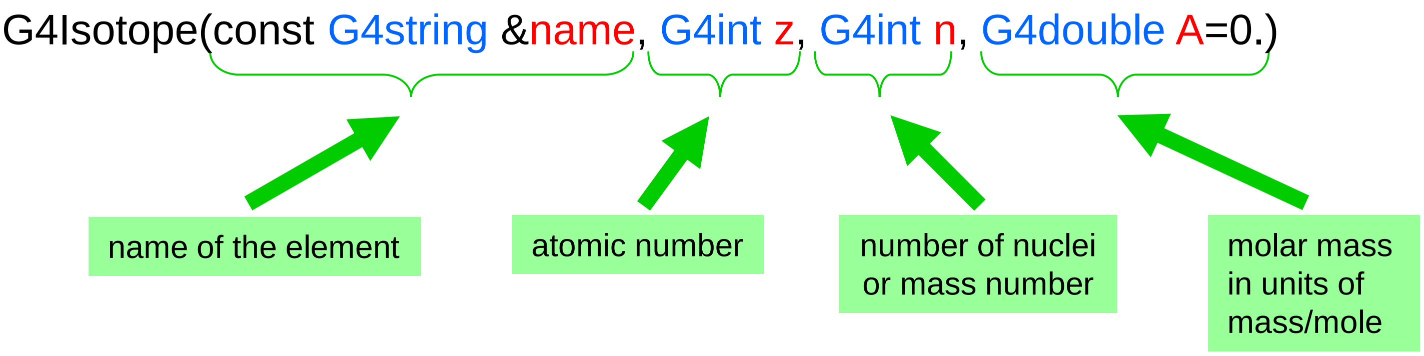
The ***G4Element*** and ***G4Isotope*** classes describe the properties of the atoms:

- atomic number,
- number of nucleons,
- atomic mass,
- as well as quantities such as cross sections per atom, etc.

The ***G4Material*** class describes the macroscopic properties of matter:

- density,
- state,
- temperature,
- pressure,
- as well as macroscopic quantities like radiation length, mean free path, dE/dx, etc.

# Isotopes



```
G4Isotope(const G4string &name, G4int z, G4int n, G4double A=0.)
```

name of the element

atomic number

number of nuclei or mass number

molar mass in units of mass/mole

*Only one way of defining an isotope!*

***Small example:***

```
G4string symbol;
G4int Z, n;
G4double A;
G4Isotope* iso_U238  = new G4Isotope("U238", Z=92, n=238, A=238.03*g/mole);
```

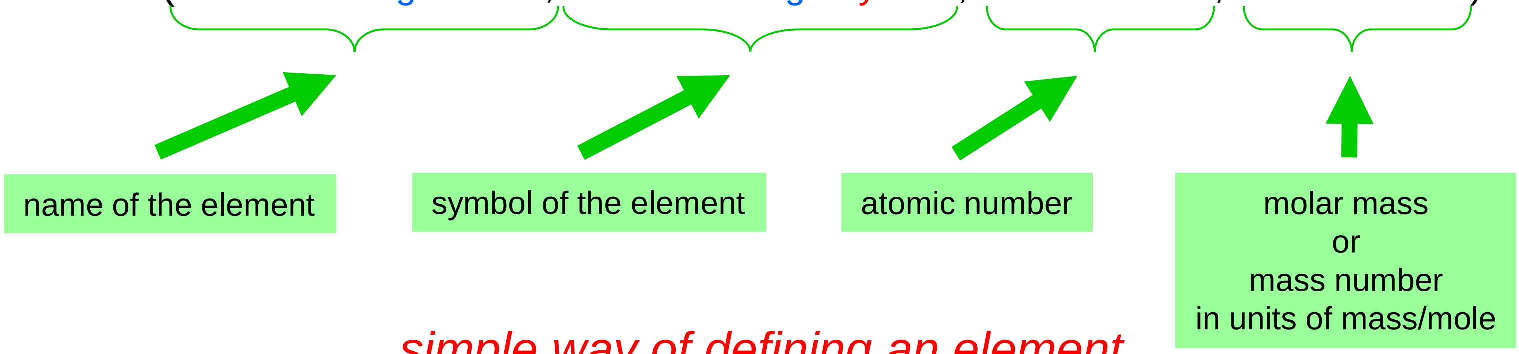
**Warning:**
This is only for better understanding.
Do not confuse it with Python's syntax!

***Usual C++ Syntax:***

```
G4Isotope* iso_U238  = new G4Isotope("U238", 92, 238, 238.03*g/mole);
```

# Elements (simple way)

```
G4Element(const G4string &name, const G4string &symbol, G4double Zeff, G4double Aeff)
```



name of the element

symbol of the element

atomic number

molar mass
or
mass number
in units of mass/mole

*simple way of defining an element*

***Small example:***

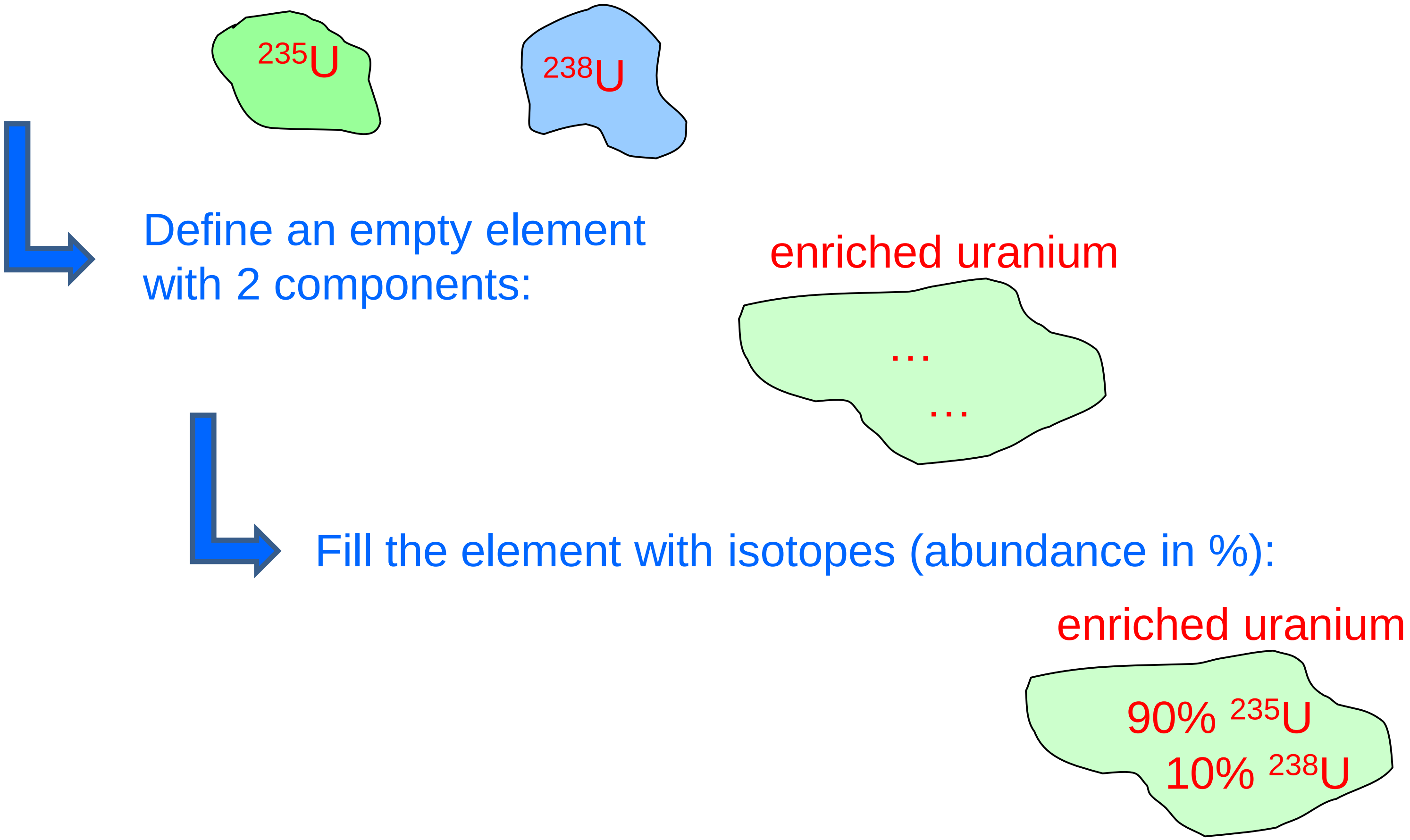
```
G4string symbol;
G4int Z;
G4double A;
G4Element* el_O  = new G4Element("Oxygen", symbol="O", Z=8, A=16.00*g/mole);
```

***Usual C++ Syntax:***

```
G4Element* el_O  = new G4Element("Oxygen", "O", 8, 16.00*g/mole);
```

# Elements with Isotopes

Define: Isotope 1 Isotope 2



$^{235}$U

$^{238}$U

Define an empty element with 2 components:

enriched uranium

…

…

Fill the element with isotopes (abundance in %):

enriched uranium

90% $^{235}$U

10% $^{238}$U

# Elements with Isotopes

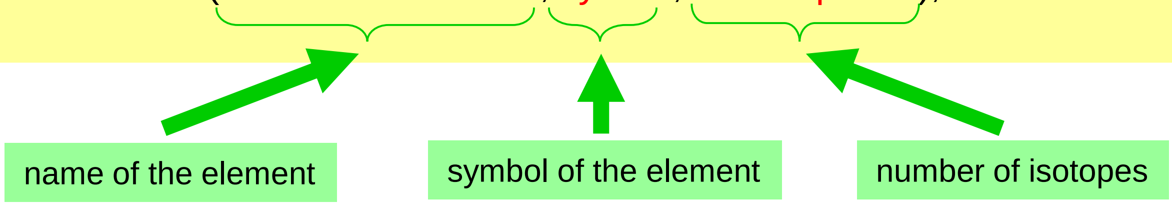
***Step 1:***

```
G4string sy; G4int Z, n; G4double A;

G4Isotope* iso_U238 = new G4Isotope(“Uranium-238”, Z=92, n=238, A=238.03*g/mole);
G4Isotope* iso_U235 = new G4Isotope(“Uranium-235”, Z=92, n=235, A=235.01*g/mole);
```

***Step 2:***

```
G4int nbIsotopes;
G4Element* el_U  = new G4Element("enriched Uranium", sy="U", nbIsotopes=2);
```



name of the element

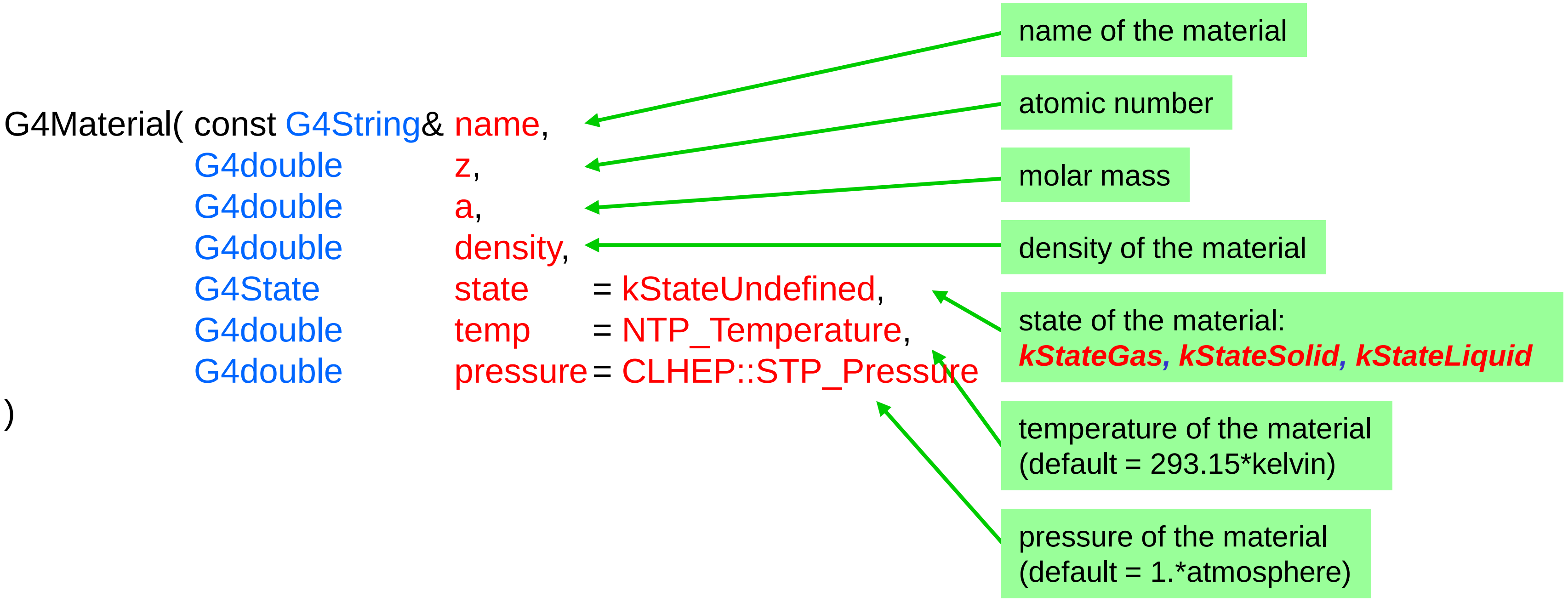
symbol of the element

number of isotopes

***Step 3:***

```
G4double abundance;
el_U->AddIsotope(iso_U235, abundance= 90.*perCent);
el_U->AddIsotope(iso_U238, abundance= 10.*perCent);
```

# Pure Materials (simple way)



```
G4Material( const G4String& name,
                  G4double       z,
                  G4double       a,
                  G4double       density,
                  G4State        state    = kStateUndefined,
                  G4double       temp     = NTP_Temperature,
                  G4double       pressure = CLHEP::STP_Pressure
)
```

name of the material

atomic number

molar mass

density of the material

state of the material:
***kStateGas***, ***kStateSolid***, ***kStateLiquid***

temperature of the material
(default = 293.15*kelvin)

pressure of the material
(default = 1.*atmosphere)

*simple way of defining a material using simple elements*

# Pure Materials (simple way)

Directly using the information of an element

```
G4double Z, A, density, T, p;

G4Material* mat_Al =  new G4Material("Aluminium",
                                     Z=13.,
                                     A=26.98*g/mole,
                                     density=2.700*g/cm3,
                                     kStateSolid,
                                     T=293.15*kelvin,
                                     p=1.*atmosphere);
```

or
simply

```
G4double Z, A, density;

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

# Materials—Mixtures

```
G4Material( const G4String& name,
                  G4double       density,
                  G4int          nComponents,
                  G4State        state     = kStateUndefined,
                  G4double       temp      = NTP_Temperature,
                  G4double       pressure  = CLHEP::STP_Pressure
)
```

name of the material

density of the material

number of components

state of the material:
***kStateGas*, *kStateSolid*, *kStateLiquid***

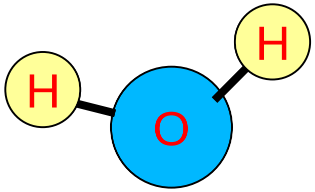
temperature of the material
(default = 293.15*kelvin)

pressure of the material
(default = 1.*atmosphere)

# Mixtures/Molecules

## Example of Water Molecule

```
G4double Z, A, density;
G4int ncomps, natoms;
G4string symbol;
```



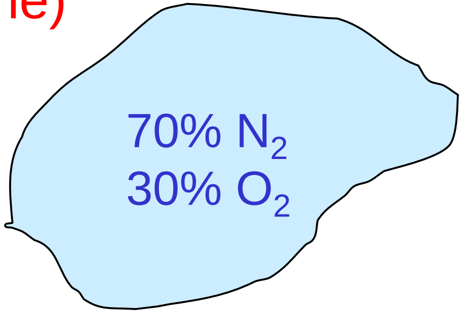
```
G4Element* el_H = new G4Element("Hydrogen", symbol="H", Z=1., A=1.01*g/mole);
G4Element* el_O = new G4Element("Oxygen",     symbol="O", Z=8., A=16.00*g/mole);
```

```
G4Material* mat_H2O = new G4Material("Water", density=1.000*g/cm3, ncomps=2);

mat_H2O->AddElement(el_H, natoms=2);
mat_H2O->AddElement(el_O, natoms=1);
```

# Mixture by Fractional Mass

## Example of Air (simple one)



70% $N_2$
30% $O_2$

```
G4double Z, A, density, fractionmass;
G4int ncomps;
G4string symbol;
```

```
G4Element* el_N = new G4Element("Nitrogen", symbol="N", Z= 7., A=14.01*g/mole);
G4Element* el_O = new G4Element("Oxygen",  symbol="O" , Z= 8., A=16.00*g/mole);
```

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

mat_Air->AddElement(el_N, fractionmass=0.7);
mat_Air->AddElement(el_O, fractionmass=0.3);
```

# Mixture of Materials and Elements

Example of Aerogel (62.5 % $SiO_2$, 37.4% $H_2O$, 0.1% C)

```
G4double density, fractionmass;
G4int ncomps;
```

```
G4Element* el_Si  = new G4Element(…);
G4Element* el_O  = new G4Element(…);
G4Element* el_H  = new G4Element(…);
G4Element* el_C  = new G4Element(…);
```

```
G4Material* mat_SiO2  = new G4Material(…);
mat_SiO2->AddElement(…); …
G4Material* mat_H2O  = new G4Material(…);
mat_H2O->AddElement(…); …
```

```
G4Material* mat_Aerog = new G4Material("Aerogel", density=0.200*g/cm3, ncomps=3);
mat_Aerog->AddMaterial(mat_SiO2,  fractionmass=62.5*perCent);
mat_Aerog->AddMaterial(mat_H2O,   fractionmass=37.4*perCent);
mat_Aerog->AddElement(el_C,        fractionmass=  0.1*perCent);
```

# Database of Materials and Elements

## Using NIST (National Institute of Standards and Technology) Database

```
#include "G4Material.hh"
#include "G4NistManager.hh"
```

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

```
// define pure NIST materials
G4Material* Al = man->FindOrBuildMaterial("G4_Al");
G4Material* Cu = man->FindOrBuildMaterial("G4_Cu");
```

```
// define NIST materials
G4Material* H2O = man->FindOrBuildMaterial("G4_WATER");
G4Material* SiO2 = man->FindOrBuildMaterial("G4_SILICON_DIOXIDE");
G4Material* Air = man->FindOrBuildMaterial("G4_AIR");
```

see chapter 11.6 (“Geant4 Material Database”) of Book for Application Developers

material from NIST database with different properties:

```
man->BuildMaterialWithNewDensity(const G4String& name, const G4String &basename,
        G4double density = 0.0, G4double temp = NTP_Temperature,
        G4double pres = CLHEP::STP_Pressure)
```

# Exercise

1. Download DetectorPhys_T5.tar.gz and decompress it.
2. Define different elements, isotopes, and materials in DetectorPhysDetectorConstruction.cc.
3. Test the effects of theses materials with:
   `/run/beamOn 100`