System of Units
Hep System of units

**GEANT4** offers the user the possibility to choose and use the units he prefers for any quantity.

Internally **GEANT4** uses a *consistent* set on units based on:

- millimeter (mm)
- nanosecond (ns)
- Mega electron Volt (MeV)
- positron charge (eplus)
- degree Kelvin (kelvin)
- the amount of substance (mole)
- luminous intensity (candela)
- radian (radian)
- steradian (steradian)
All the others units are defined from the basic ones. For instance:

\[
\text{millimeter} = \text{mm} = 1. ; \\
\text{meter} = \text{m} = 1000.*\text{mm}; \\
\text{meter3} = \text{m3} = \text{m*mm*mm}; \\
\ldots \\
\text{nanosecond} = \text{ns} = 1. ; \\
\text{second} = \text{s} = 1.e+9*\text{ns}; \\
\text{hertz} = 1./\text{second}; \\
\ldots
\]

See: geant4/source/global/management/include/SystemOfUnits.h

This file is also part of CLHEP
Input your data

You must give the units for the data you are going to introduce:
G4double Size = 15*km, Energy = 90.3*GeV;
G4double density = 11*mg/cm3;

If the unit is not specified, it is understood that the data is implicitly in the internal G4 system, but this is discouraged.

If the data set comes from an array or from an external file, it is recommended to set the units as soon as the data are read, before any treatment. For instance, immediately after to have read a file of cross sections expressed in millibarn:
for (int j=0, j<jmax, j++) CrossSection[j] *= millibarn;

..........................                       
....... your calculations ...

Scant4 Tutorial Slac, 18.02.02
Interactive commands

Some built-in commands from the User Interface (UI) also require the unit.

For instance:
/gun/energy 15.2 keV
/gun/position 3 2 -7 meter

If the unit is not specified or not valid the command is refused.
Output your data

You can output the data on the units you wish. To do so it is sufficient to divide the data by the corresponding unit:

G4cout << KineticEnergy /keV << "keV" ;
G4cout << density /(g/cm3) << "g/cm3" ;

Of course, G4cout << KineticEnergy
will print the energy in the internal system of units.
There is another way to output the data; let Geant4 choose the most appropriate unit to the actual numerical value of your data. It is sufficient to specify to which category your data belong (Length, Time, Energy ..etc..)

for example:

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

StepSize will be printed in km, m, mm or ... fermi depending of its actual value.
Introduce new units

If you wish to introduce new units, there are two methods:
you can complete the file SystemOfUnits.h:
#include "SystemOfUnits.h"
static const G4double Inch = 2.54*cm;
static const G4double Ounce = 28.35*g;
In this way it is not easy to define composed units.
It is better to do the following:
Instantiate an object of the class `G4UnitDefinition`:

```
G4UnitDefinition ( name, symbol, category, value)
```

**example**: introduce few units for mass-thickness:

```
G4UnitDefinition ("grammepercm2",
    "g/cm2","MassThickness",g/cm2);
G4UnitDefinition ("kilogrammeperm2",
    "kg/m2","MassThickness",kg/m2);
```

The category `MassThickness` does not exist by default in `G4UnitsTable`, but it will be created automatically.

The classes `G4UnitDefinition` and `G4UnitsTable` are located in:
`geant4/source/global/management`
Change the internal system of units

We can change the system of units used internally by GEANT4. One must redefine the basic units in SystemOfUnits.h and recompile the whole GEANT4 kernel, since the units are heavily used elsewhere (especially in the physics sector).

Indeed the Geant4 code is written respecting the above conventions and this makes it independent of the units chosen.

The file:
source/global/management/include/PhysicalConstants.h illustrates how the physics is coded. It is recommended to use it.
5 GeV shower profiles: independence from the system of units

PbWO4  e- 5 GeV  G4 System Of Units

(longit energy profile (% of E inc)

radial energy profile (% of E inc)

(default units)

(SI units)
Print the list of units

From code: G4UnitDefinition::PrintUnitsTable();

With UI command: Idle> /units/list

More examples can be found in:
source/global/management/test/G4UnitsTableTest.cc