

Geant4: Geometry Basics B

Boolean Operations

Andreas Nowack

nowack@physik.rwth-aachen.de

RWTH Aachen University

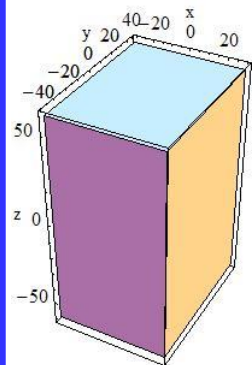
WS 2020/21



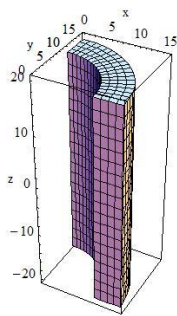
How Many Predefined CSG Do We Have?



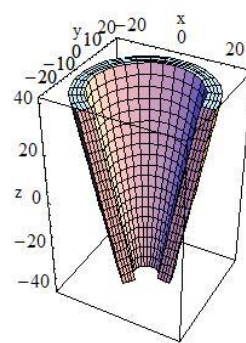
Recap



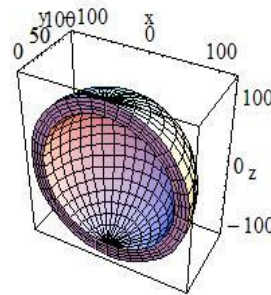
Box



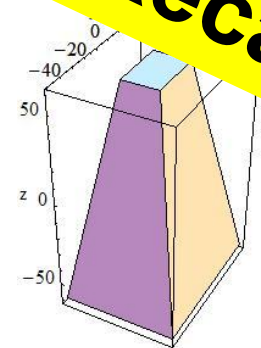
Tube



Cone

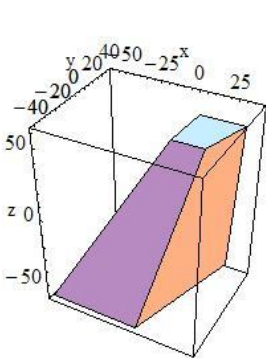


Sphere

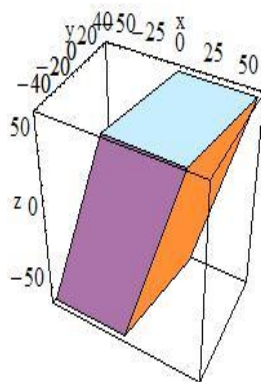


Trapezoid

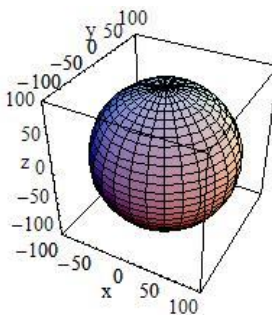
My preferred ones... very basic and simple... see exercise of last week



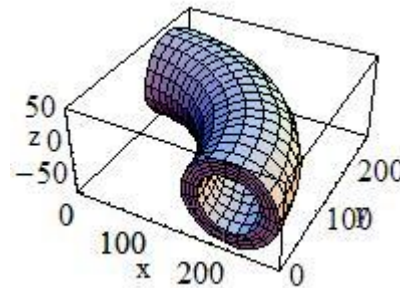
Generic
Trapezoid



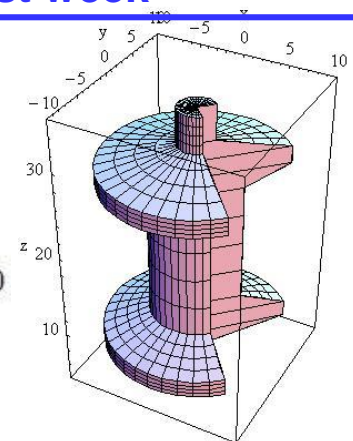
Parallelepiped



Solid
Sphere



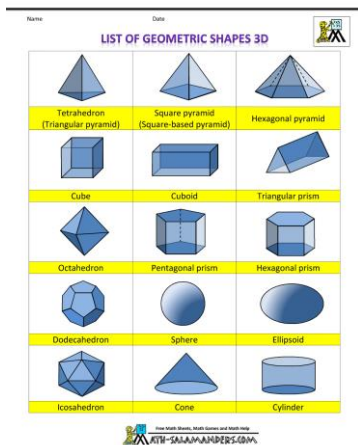
Torus



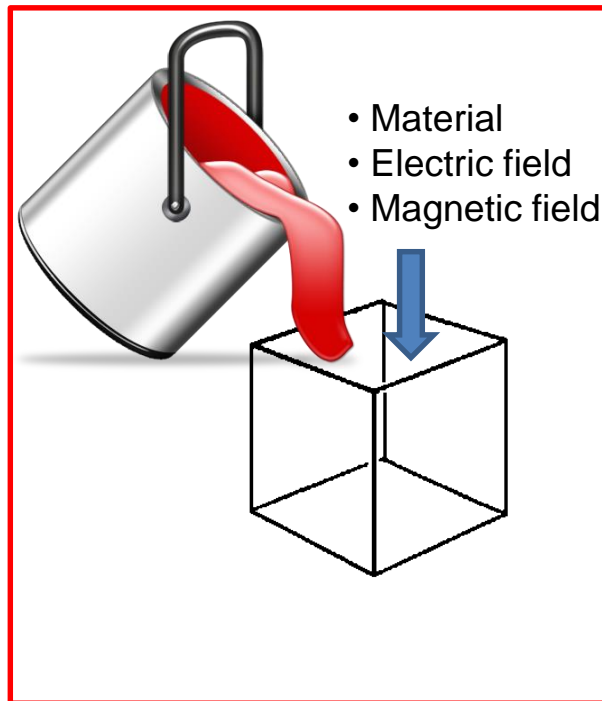
Polycons

Geometry in Three Steps

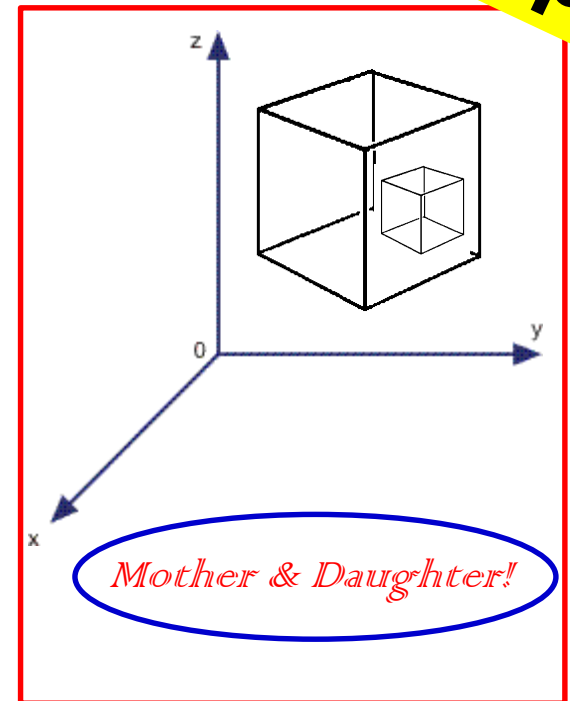
Recap



Mathematical shape
(Solid)



Logical Volume



Placement in (X, Y, Z)
Physical Volume

Geometry in Three Steps

Recap

```
G4Tubs* SolidMyCylinder = new G4Tubs("SolidMyCylinder",  
                                       Rmin ,  
                                       Rmax,  
                                       Lc/2. ,  
                                       PhiMin1,  
                                       PhiMax1);
```

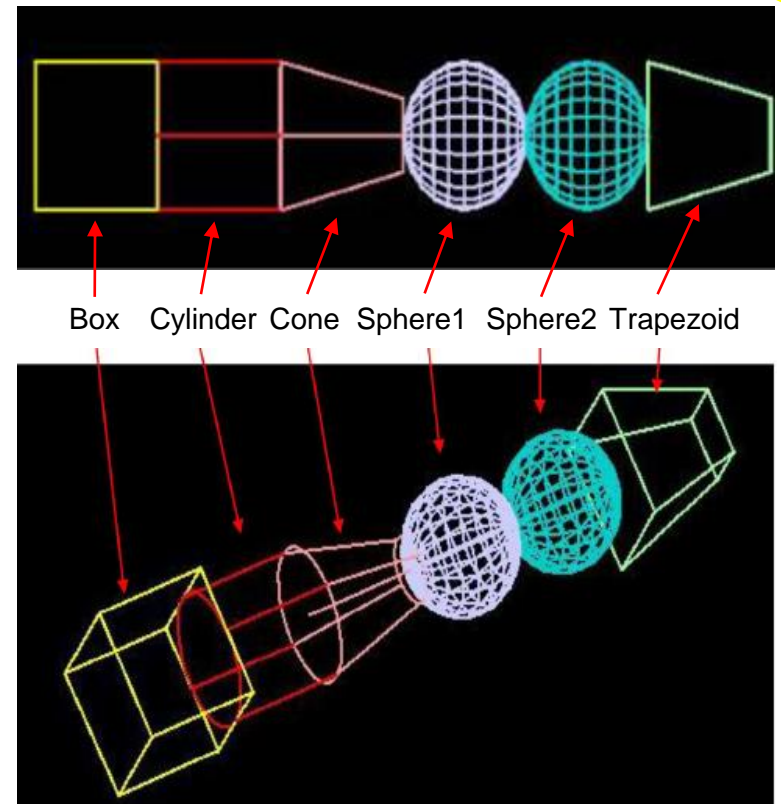
```
G4LogicalVolume LogicalMyCylinder = new G4LogicalVolume(SolidMyCylinder, //its solid  
                                                         Vacuum,           //its material  
                                                         "LogicalMyCylinder"); //its name
```

```
G4VPhysicalVolume* PhysicalMyCylinder = new G4PVPlacement(0, //no rotation  
                                                         G4ThreeVector(0,0,0), //at (0,0,0)  
                                                         "PhysicalMyCylinder", //its name  
                                                         LogicalMyCylinder, //its logical volume  
                                                         physWorld, //its mother volume  
                                                         false, //no Boolean operation  
                                                         0); //copy number
```

Exercise of Last Week

Recap

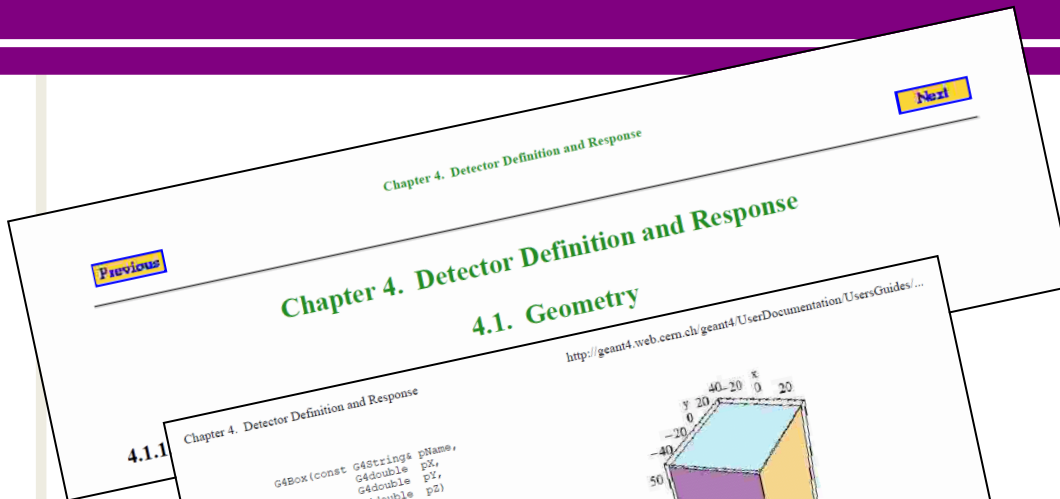
- adjacent volumes sharing the same mother volume “world”
- always check the geometry:
Session:
`/geometry/test/run`



Where to Find the CSG (Predefined Geometry)

Recap

Geant4 Documentation

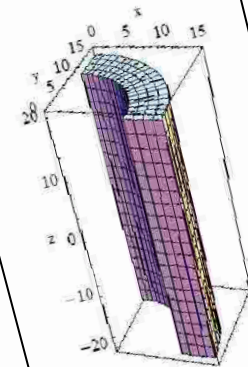


by giving the box a name and its half-lengths along the X, Y and Z axis:

pX | half length in X | pY | half length in Y | pZ | half length in Z

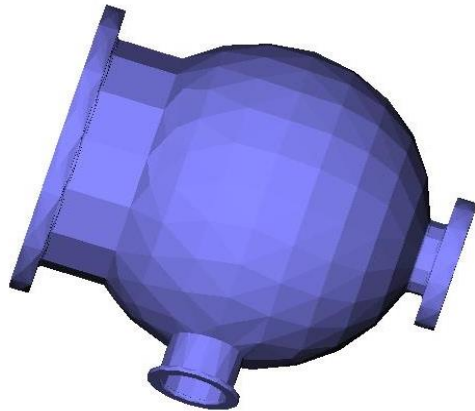
Cylindrical Section or Tube:
Similarly to create a cylindrical section or tube, one would use the constructor:

```
G4Tubs(const G4String& pName,  
       G4double pRMin,  
       G4double pRMax,  
       G4double pDz,  
       G4double pSPhi,  
       G4double pDPhi)
```



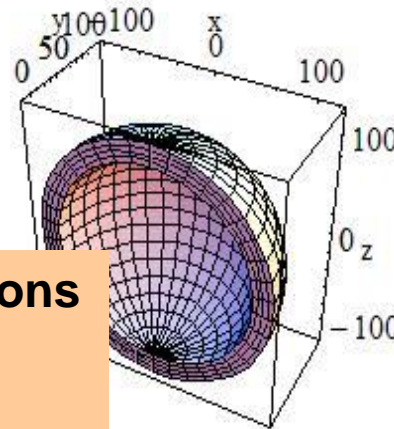
[\[Rotate the Picture\]](#)
In the picture :

What About a Special Geometry?

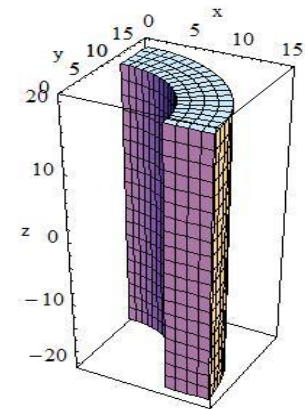


For example this one...

This is a result of
Boolean operations on CSGs.



Sphere



Tube

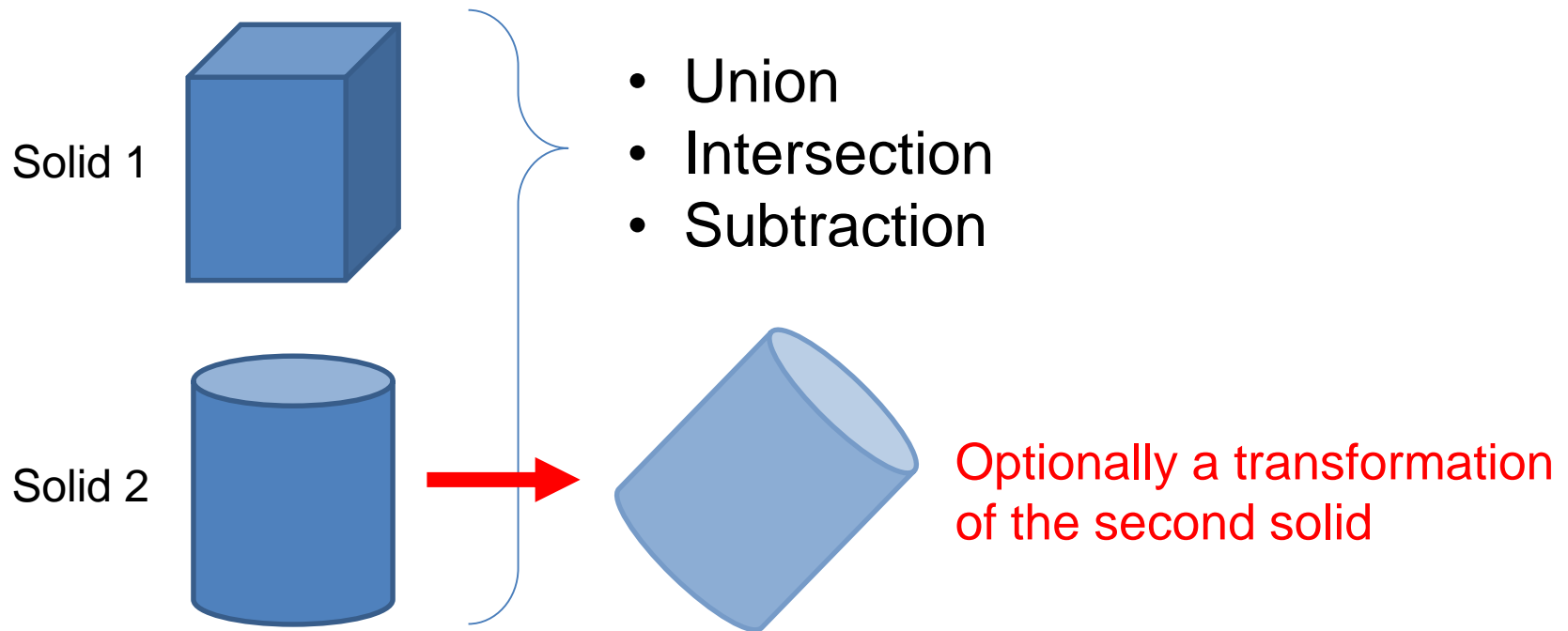
We have only three Boolean operations

1. Subtractions
2. Unions
3. Intersections



Boolean Operations

- Simple solids can be combined using Boolean operations.
- For example, a cylinder and a cube can be combined with each other using:



Boolean Operations

- Depending on which Boolean operation you want to perform do not forget to include the following header files:
 - G4UnionSolid.hh
 - G4IntersectionSolid.hh
 - G4SubtractionSolid.hh

```
// ooooOOOOOOOOoooo Solid Box

G4double BoxX = ---- *cm;
G4double BoxY = ---- *cm;
G4double BoxZ = ---- *cm;

G4Box * SolidBox = new G4Box("SolidChamberGasBox", //its name
                             BoxX/2,BoxY/2,BoxZ/2); //its size
```

Step 1: define solid 1

```
// ooooOOOOOOOOoooo Solid Cylinder

G4double CylinderRmin = ---- *cm;
G4double CylinderRmax = ---- *cm;
G4double CylinderZ = ---- *cm;
G4double CylinderPhiMin = ---- *deg;
G4double CylinderPhiMax = ---- *deg;

G4Tubs* SolidCylinder = new G4Tubs("SolidCylinder",
                                   CylinderRmin,
                                   CylinderRmax,
                                   CylinderZ,
                                   CylinderPhiMin,
                                   CylinderPhiMax
                                   );
```

Step 2: define solid 2

Boolean Operations

- Define a 3D transformation which will be applied to **solid 2**
 - one 3D rotation
 - one 3D translation

Step 3: 3D transform

Don't forget
G4RotationMatrix.hh

```
G4RotationMatrix* My3DRotation = new G4RotationMatrix;  
My3DRotation->rotateX(----*rad);  
My3DRotation->rotateY(----*rad);  
My3DRotation->rotateZ(----*rad);  
  
G4ThreeVector My3DTranslation(----*cm, ----*cm, ----*cm);
```

Boolean Operations

Step 4: Union

```
// Union Box Cylinder  
G4UnionSolid* BoxCylinderUnion = new G4UnionSolid("Box+Cylinder", SolidBox, SolidCylinder, My3DRotation, My3DTranslation );
```

solid 1

solid 2

Rotation + Translation
(3D Transformation)

```
G4LogicalVolume* LogicalBoxCylinderUnion = new G4LogicalVolume(BoxCylinderUnion, //its solid  
                                                             Vacuum, //defaultMaterial,  
                                                             "LogicalBoxCylinderUnion"); //its name
```

```
G4VPhysicalVolume* PhysicalBoxCylinderUnion = new G4PVPlacement(0, //no rotation  
                                                             G4ThreeVector(), //at (0,0,0)  
                                                             "PhysicalBoxCylinderUnion", //its name  
                                                             LogicalBoxCylinderUnion, //its logical volume  
                                                             physiWorld, //its mother volume  
                                                             false, //no boolean operation  
                                                             0); //copy number
```

Boolean Operations

Intersection

```
// Intersection Box Cylinder  
G4IntersectionSolid* BoxCylinderIntersection = new G4IntersectionSolid("Box*Cylinder", SolidBox, SolidCylinder, My3DRotation, My3DTranslation);
```

solid 1

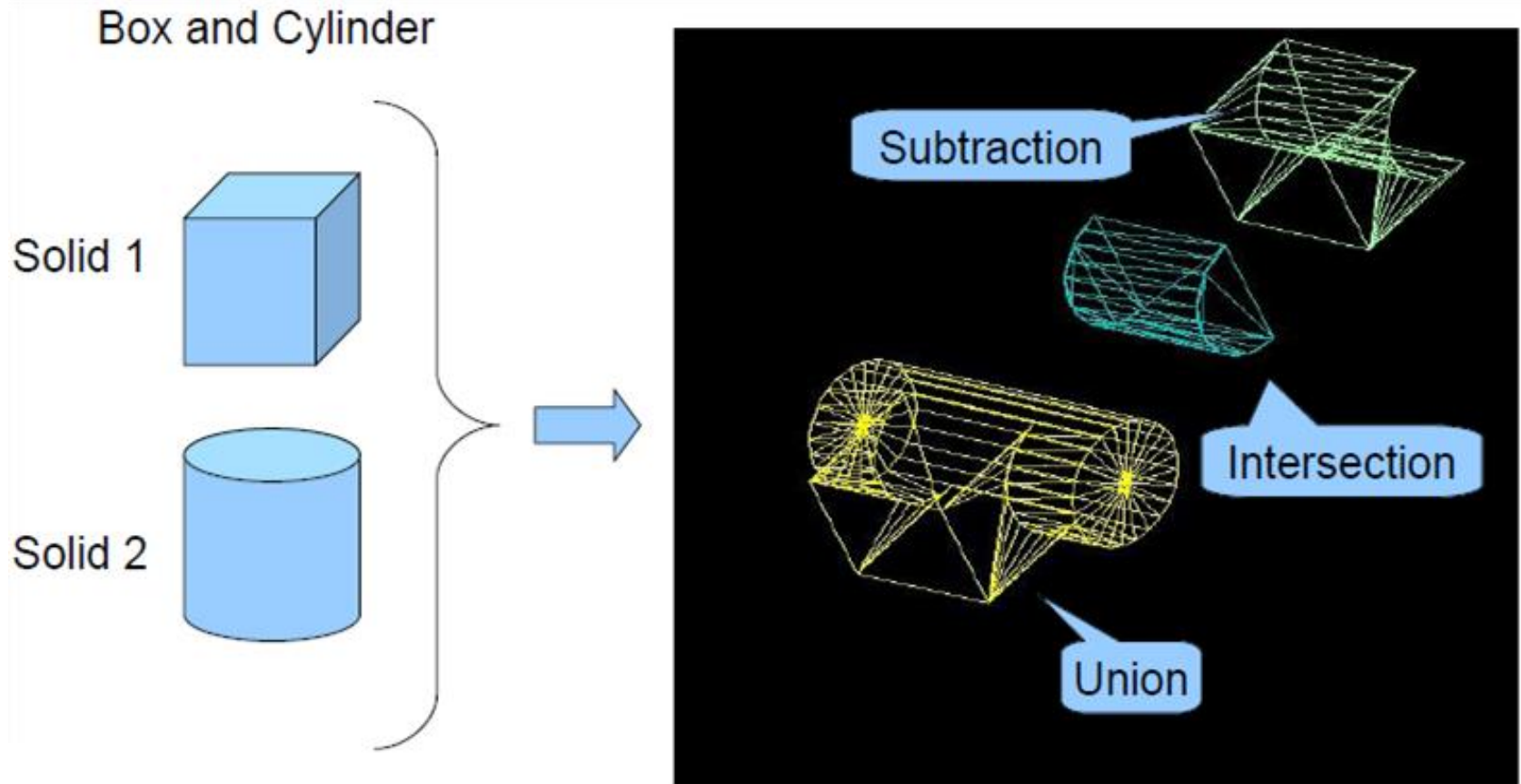
solid 2

Rotation + Translation
(3D Transformation)

Subtraction

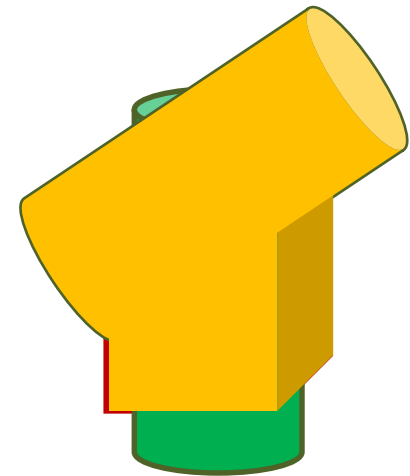
```
// Subtraction Box Cylinder  
G4SubtractionSolid* BoxCylinderSubtraction = new G4SubtractionSolid("Box*Cylinder", SolidBox, SolidCylinder, My3DRotation, My3DTranslation);
```

Solids Made by Boolean Operations



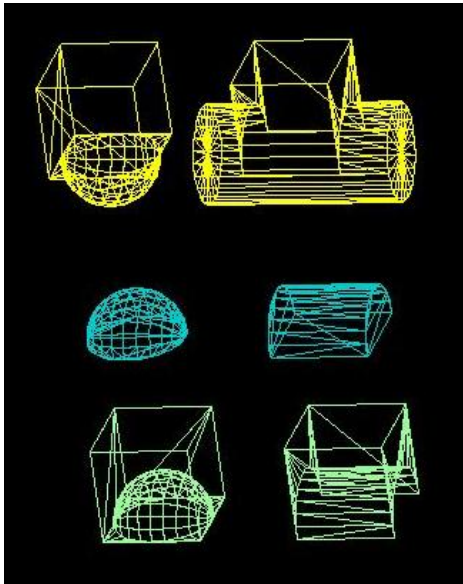
Composing Solids Step by Step

1. Define only **solid 1**
(mathematical shape of predefined CSG or previous Boolean operation)
You **do not need** logical volume and placement!
2. Define only **solid 2**
(mathematical shape of predefined CSG or previous Boolean operation)
You **do not need** logical volume and placement!
3. **Translation** and **Rotation** of **solid 2**
4. Boolean operation to make a **new solid**
5. **Logical Volume** for the **new solid**
6. **Physical Volume (placement)** for the **new solid**



Exercise 1

1. Download [DetectorPhys_T2.tar.gz](#) and decompress it.
2. Edit [DetectorPhysDetectorConstruction.cc](#) and follow the instructions there.
3. Check your geometry with:
`/geometry/test/run`



Exercise 2

1. How many and which solids have to be defined?
2. How many Boolean operations are needed?
And which ones?

