

# EM Processes and Tracking

V.Ivanchenko  
CERN, EMSU

11<sup>th</sup> Geant4 Workshop, 12-19 September  
Hebden Bridge, Uk

# Motivation

- ▶ To achieve higher CPU performance of simulation with maximum precision of physics results EM models are using geometry information
  - G4UrbanMscModel
  - G4VEnergyLossProcess for sub-cutoff option
- ▶ Since 9.0 only G4SafetyHelper is used not G4Navigator
  - Parallel navigation enabled

# Where we use geometry/navigation?

- ▶ G4UrbanMscModel:
  - GetContinuousStepLimit
  - PostStepDoIt
- ▶ G4VEnergyLossProcess
  - AlongStepDoIt
- ▶ We are using following methods:
  - Call to G4SafetyHelper – unique object for all processes
  - Call to G4StepPoint
- ▶ Below all calls will be shown in chronological order

# G4UrbanMscModel:: ComputeTruePathLengthLimit

## ► for any option

```
preSafety = preStepPoint->GetSafety();  
if (currentRange < preSafety) {  
    inside = true;  
    return tPathLength;  
}
```

## ► default, fUseSafety, skin = 0

```
if (stepStatus != fGeomBoundary && presafety < tlimitminfix) {  
    preSafety = safetyHelper->ComputeSafety(pos);  
}
```

# G4UrbanMscModel:: ComputeTruePathLengthLimit

► fDistanceToBoundary, skin > 0

```
if (track->GetVolume() != 0 &&  
    track->GetVolume() != safetyHelper->GetWorldVolume())  
{  
    geomlimit = safetyHelper->CheckNextVolume(pos, dir,  
        currentRange, preSafety);  
}  
// preSafety can be modified (?!)
```

# G4Transportation

- ▶ GetContinuousStepLimit
- ▶ AlongStepDoIt

# G4VEnergyLossProcess:: AlongStepDoIt

► Only if sub-cutoff option is enabled

```
preSafety = preStepPoint->GetSafety();
```

```
postSafety = preSafety - length;
```

```
currentMinSafety = 0.;
```

```
if (prePoint->GetStepStatus() != fGeometryBoundary &&  
    postPoint->GetStepStatus() != fGeometryBoundary ) {
```

```
    if(preSafety < rcut) preSafety = safetyHelper->GetSafety(prePos);
```

```
    if(postSafety < rcut) postSafety = safetyHelper->GetSafety(postPos);
```

```
    currentMinSafety = min(preSafety, postSafety);
```

```
}
```

```
if(currentMinSafety < rcut) {
```

```
    //Sampling sub-cutoff secondary between pre and post step points
```

```
    if(range < safetyHelper->ComputeSafety(position)) t = new G4Track;
```

```
}
```

# G4UrbanMscModel:: PostStepDoIt for any option

```
safety = postStepPoint->GetSafety();
if(latDisplacement && safety > tlimitminfix) {
    // sample lateral displacement delta
    fac = 1.;
    if(delta > safety) {
        newsafety = safetyHelper->ComputeSafety(position);
        if(delta > newsafety) fac = newsafety/delta;
    }
    newPosition = position + fac*delta*direction;
    if(1. == fac) safetyHelper->RelocateWithinVolume(newPosition);
    else {
        postSafety = safetyHelper->ComputeSafety(newPosition);
        if(postSafety <= 0) safetyHelper->Locate(newPosition, newDirection);
        else
            safetyHelper->RelocateWithinVolume(newPosition);
    }
}
```



# Comments/questions

- ▶ Are G4SafetyHelper and G4StepPoints coherent?
- ▶ We still do not change PostStep volume for scattering back from the boundary

