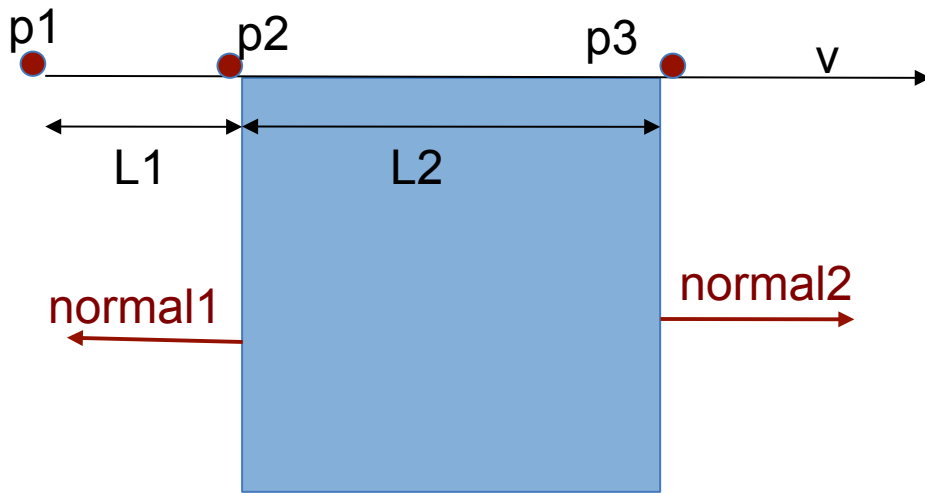


# Shapes Conventions

*Distance & Safety*

# Convex shape with flat surfaces, conventions for 'scratching' directions

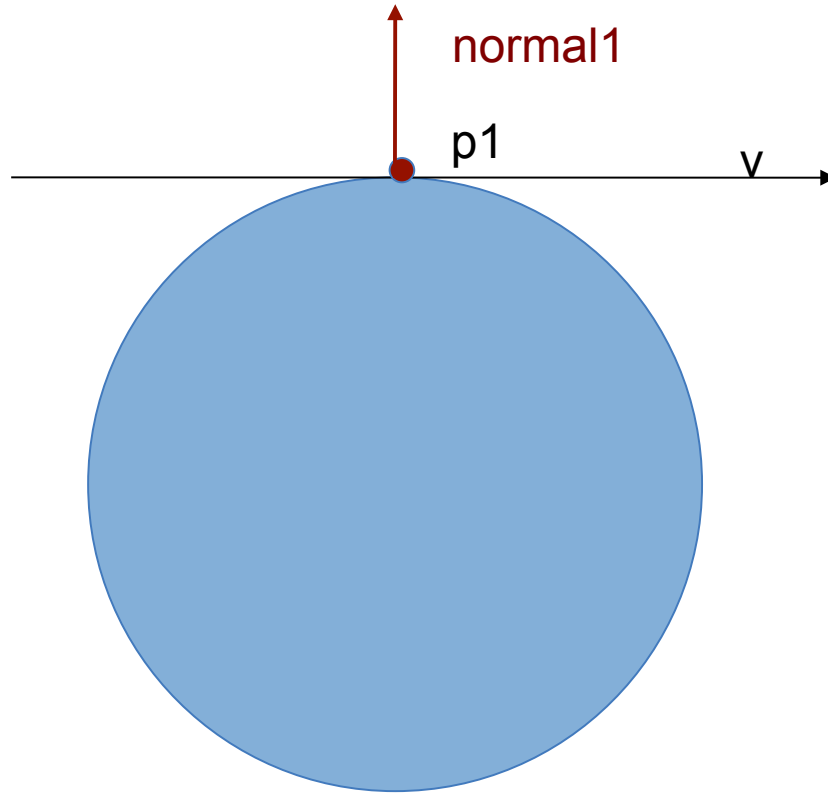


p1 : DistanceToIn(p1,v) = L1

p2 : DistanceToIn (p2,v) = 0, (normal1,v) < 0  
DistanceToOut(p2,v) = L2, (normal1,v) < 0

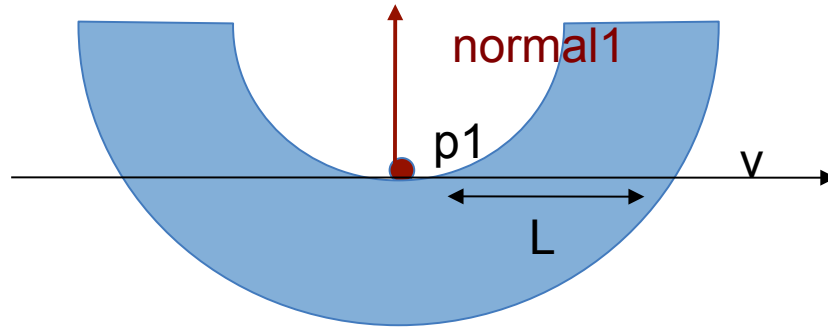
p3 : DistanceToIn (p3,v) = Infinity, (normal2,v) > 0  
DistanceToOut(p3,v) = 0, (normal2,v) > 0

# Convex shape with curved surfaces, conventions for 'scratching' directions



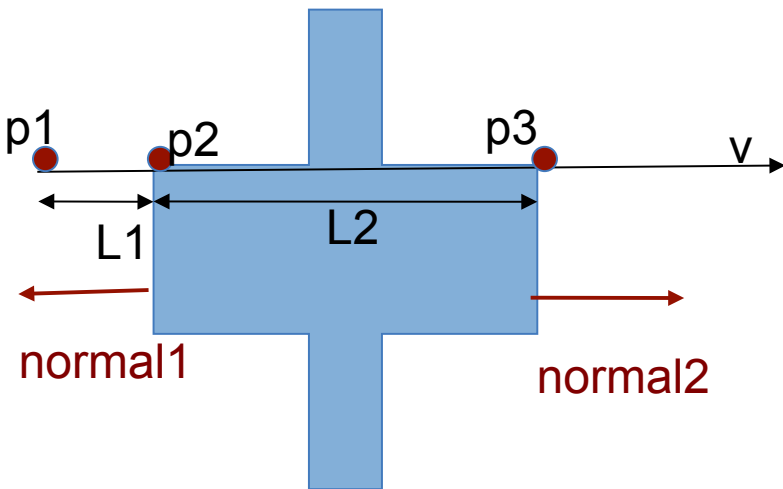
p1 : DistanceToIn (p1,v) = Infinity, (normal1,v) = 0  
DistanceToOut(p1,v) = 0, (normal1,v) = 0

## Concave shape with curved surfaces, 'scratching' directions



P1 : DistanceToIn (p1,v) = 0, (normal1,v) = 0  
 DistanceToOut(p1,v) = L, (normal1,v) = 0

## Concave shape with flat surfaces, 'scratching' directions

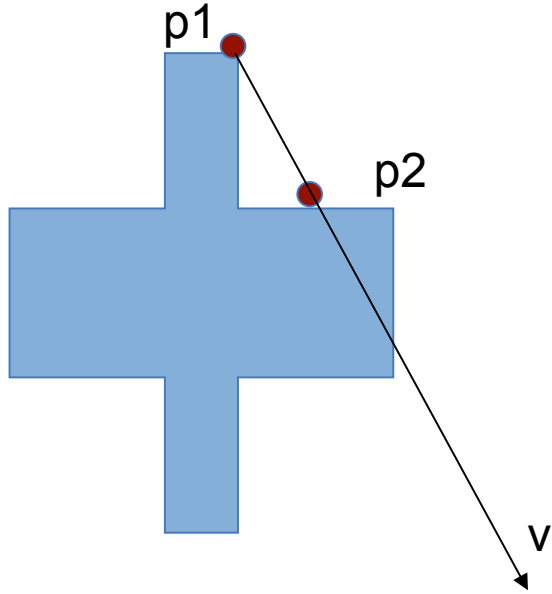


p1 : DistanceToIn(p1,v) = L1, not Infinity

p2 : DistanceToIn (p2,v) = 0, (normal1,v) < 0  
 DistanceToOut(p2,v) = L2, (normal1,v) < 0

p3 : DistanceToIn (p3,v) = Infinity, (normal2,v) > 0  
 DistanceToOut(p3,v) = 0, (normal2,v) > 0

## Concave shapes , 'reentering' directions



$p1$  :  $\text{DistanceToIn}(p1, v) = L1$ , not Infinity

# Conventions. DistanceToIn(p,v)

DistanceToIn(p,v) return the exact distance (double) to the surface of the shape for given point p and direction v. Normal is pointing outwards shape.

	Geant4	USolids	VecGeom	ROOT
Point p is <b>Outside</b> No intersection Between shape and ray(p+v*t)	Infinity	Infinity	Infinity	Infinity
Point p is <b>Outside</b> Intersection Between shape and ray(p+v*t)	Distance	Distance	Distance	Distance
Point p is <b>on Surface</b>	<b>0</b> , if 'entering shape', normal.dot(v) < 0 for convex shapes  <b>Infinity for convex shapes or Distance to next Intersection for concave shapes</b> , if 'leaving shape', normal.dot(v) >= 0 for convex shapes	<b>0</b> , if 'entering shape', normal.dot(v) < 0 for convex shapes  <b>Infinity for convex shapes or Distance to next Intersection for concave shapes</b> , if 'leaving shape', normal.dot(v) >= 0, for convex shapes	<b>0</b> , if 'entering shape'  <b>Infinity for convex shapes or Distance to next Intersection for concave shapes</b> , if 'leaving shape'	<b>0</b> , if 'entering shape', normal.dot(v) < 0 for convex shapes  <b>Infinity for convex shapes or Distance to next Intersection for concave shapes</b> , if 'leaving shape', normal.dot(v) >= 0 for convex shapes
Point p is <b>Inside</b> “Wrong side”	No general rule(?), depends on shape	0	-Distance	
If Distance(p,v)<halfTolerance	0	0	Distance	

# Conventions. DistanceToOut(p,v)

DistanceToOut(p,v) return the exact distance (double) to the surface of the shape for given point p and direction v. Normal is pointing outwards shape.

	Geant4	USolids	VecGeom	ROOT
Point p is <b>Inside</b> No intersection Between shape and ray(p+v*t), “wrong result”	Infinity, as default value	Infinity, as default value	Infinity, as default value	Infinity, as default value
Point p is <b>Inside</b> Intersection Between shape and ray(p+v*t)	Distance	Distance	Distance	Distance
Point p is <b>on Surface</b>	0, if 'leaving' shape, normal.dot(v) >= 0 for convex shape <b>DistanceTo next boundary</b> , if 'entering' shape, normal.dot(v) < 0 for convex shape	0, if 'leaving' shape, normal.dot(v) >= 0, for convex shape <b>DistanceTo next boundary</b> , if 'entering' shape, normal.dot(v) < 0 for convex shaper	0, if 'leaving shape' <b>DistanceTo next boundary</b> , if 'entering shape'	0, if 'leaving' shape, normal.dot(v) >= 0 for convex shape <b>DistanceTo next boundary</b> , if 'entering' shape ,normal.dot(v) < 0 for convex shape
Point p is <b>Outside</b> “Wrong side”	No general rule(?), depends on shape	0	-Distance	
If Distance(p,v)<halfTolerance	0	0	Distance	

# Conventions. *SafetyFromOutside(p)*

SafetyFromOutside estimates isotropic distance to the surface of the shape from Outside. This must be either accurate or an underestimate.

	Geant4	USolids	VecGeom	ROOT
Point p is <b>Outside</b>	Safety	Safety	Safety	Safety
Point p is <b>on Surface</b>	0	0	0	0
Point p is <b>Inside</b> “Wrong side”	0	0	-Safety	-Safety



# Conventions. *SafetyFromInside(p)*

*SafetyFromInside(p)* estimates isotropic distance to the surface of the shape from Inside point p. This must be either accurate or an underestimate.

	Geant4	USolids	VecGeom	ROOT
Point p is <b>Inside</b>	Safety	Safety	Safety	Safety
Point p is <b>on Surface</b>	0	0	0	0
Point p is <b>Outside</b> “Wrong side”	0	0	-Safety	-Safety