Visualizing Electromagnetic Fields: The Visualization Toolkit

Michael Selvanayagam

Visualization

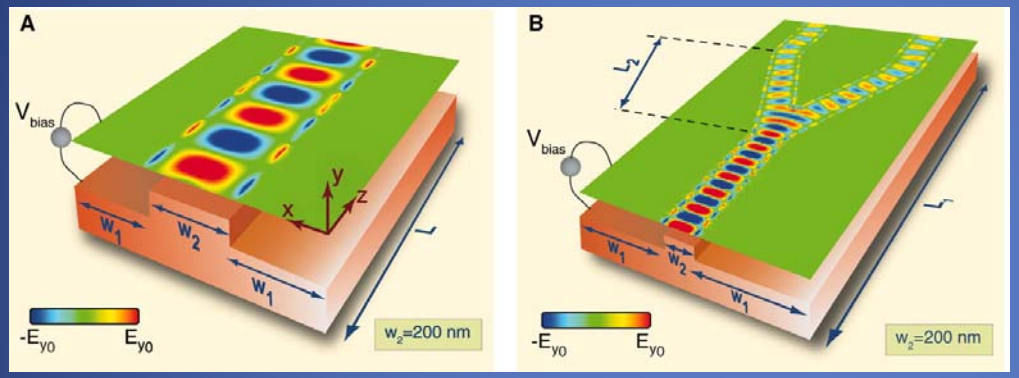
What is the purpose of visualizing electromagnetic (EM) Fields?

Visualization

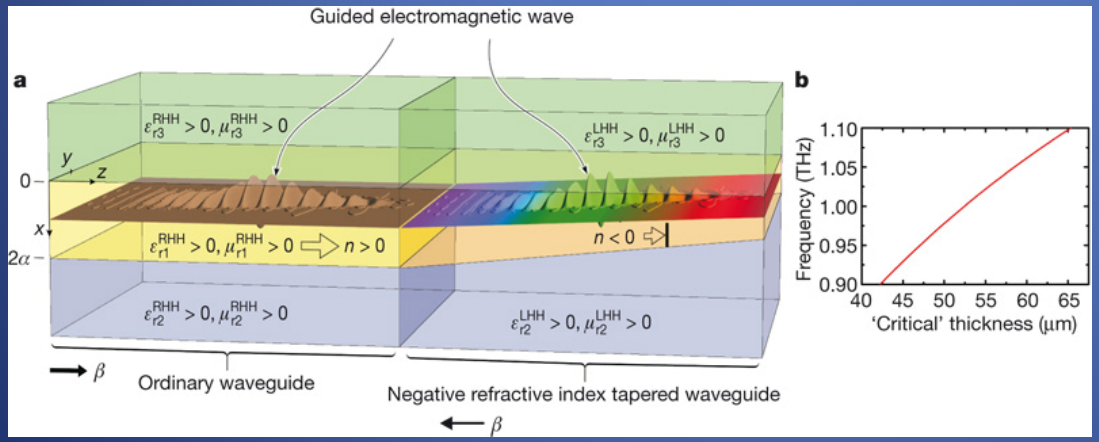
1. Understand the geometry of the problem
2. Show the properties of the EM field
3. Communicate functionality

Examples

- Good Visualization:



Transformation Optics Using Graphene, Ashkan Vakil and Nader Engheta
Science 10 June 2011: 332 (6035), 1291-1294.



Kosmas L. Tsakmakidis, Allan D. Boardman & Ortwin Hess
Nature **450**, 397-401
(15 November 2007)

Motivation

- Generate ‘professional’ looking figures
- Convey both geometry and electromagnetic properties
  - This allows the audience to understand how things work in an easier fashion

Motivation

- Useful for Research:
  - Help yourself to understand the problem
  - Help the audience to understand the problem
- Useful for Teaching:
  - Demonstrate basic physics in a visually appealing manner

Possible Approaches

- Existing CAD tools:
  - HFSS, SEMCAD, CST, COMSOL….
- Software packages

Visualization Toolkit (VTK)

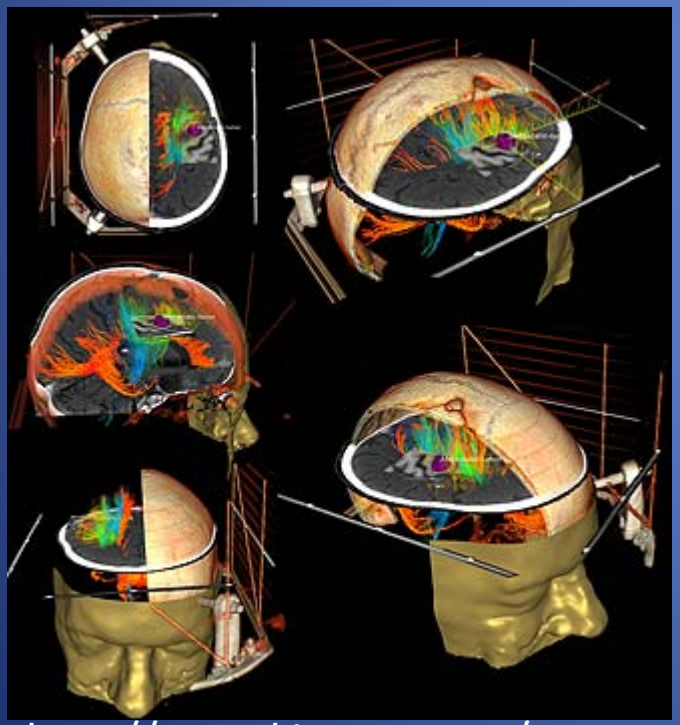
- A Graphics API for drawing 3D graphics
- Open source
- Use the API in C++, Java, Tcl, Python
  - Python in this talk. Also allows use of numpy/scipy

Visualization Toolkit (VTK)

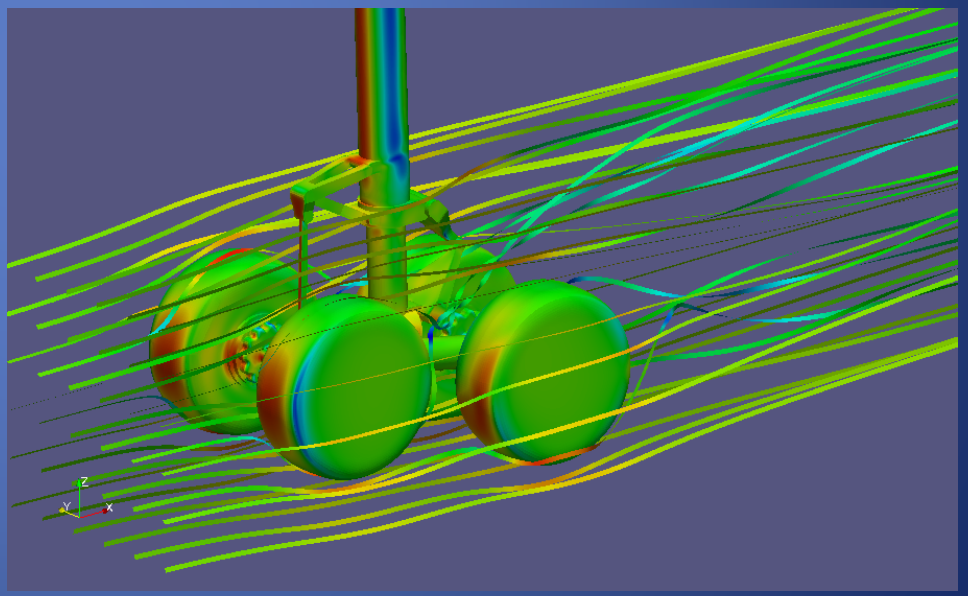
- Object Oriented Framework
- Sits on top of OpenGL
- There are GUI environments:
  - ParaView, Mayavi

VTK History and Uses

- Written by former GE engineers
- Popular in CFD and medical imaging



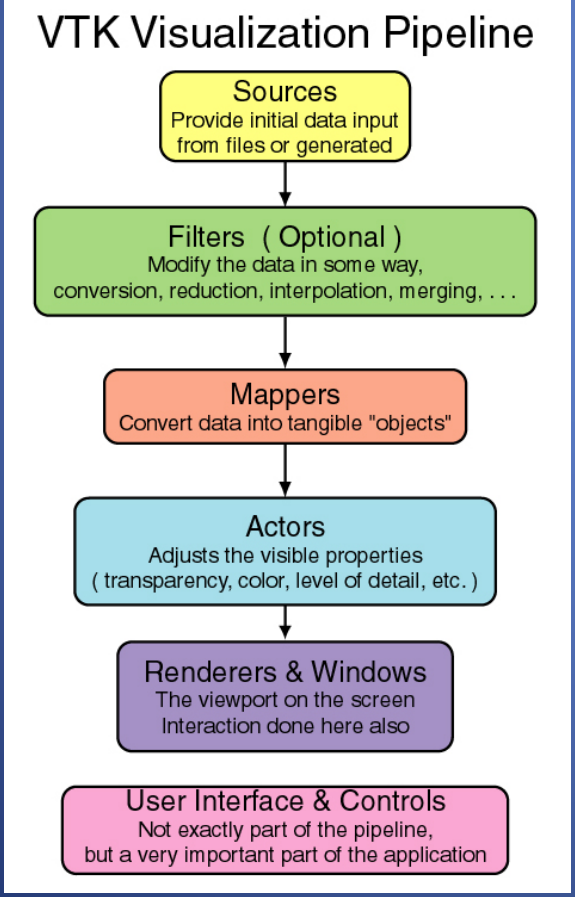
http://www.kitware.com/viscontest/img/brainconnectivity.png



http://cmg.soton.ac.uk/people/kvm/

VTK Basics

- The elements of a VTK program



Pipeline Metaphor

From: http://www.cs.uic.edu/…
~jbell/CS526/Tutorial/VTK_Pipeline.jpg

VTK Basics

- To Build A ‘Scene’ You Need:
  1. A Rendering Window (vtkRenderWindow)
  2. A Light Source (vtkLight)
  3. A Camera (vtkCamera)
  4. Mappers (vtkMapper)
     - Map data and geometry to computer graphics
  5. Actors (vtkActor)
     - Define their properties (vtkProperty)

Examples (Photonics)

- The Hybrid Waveguide:
  - An optical waveguide with good mode confinement
  - A hybrid between a dielectric and plasmonic waveguide
  - See [1].

[1] M. Alam, J. Meier, J. Aitchison, and M. Mojahedi, "Propagation characteristics of hybrid modes supported by metal-low-high index waveguides and bends," Opt. Express 18, 12971-12979 (2010).

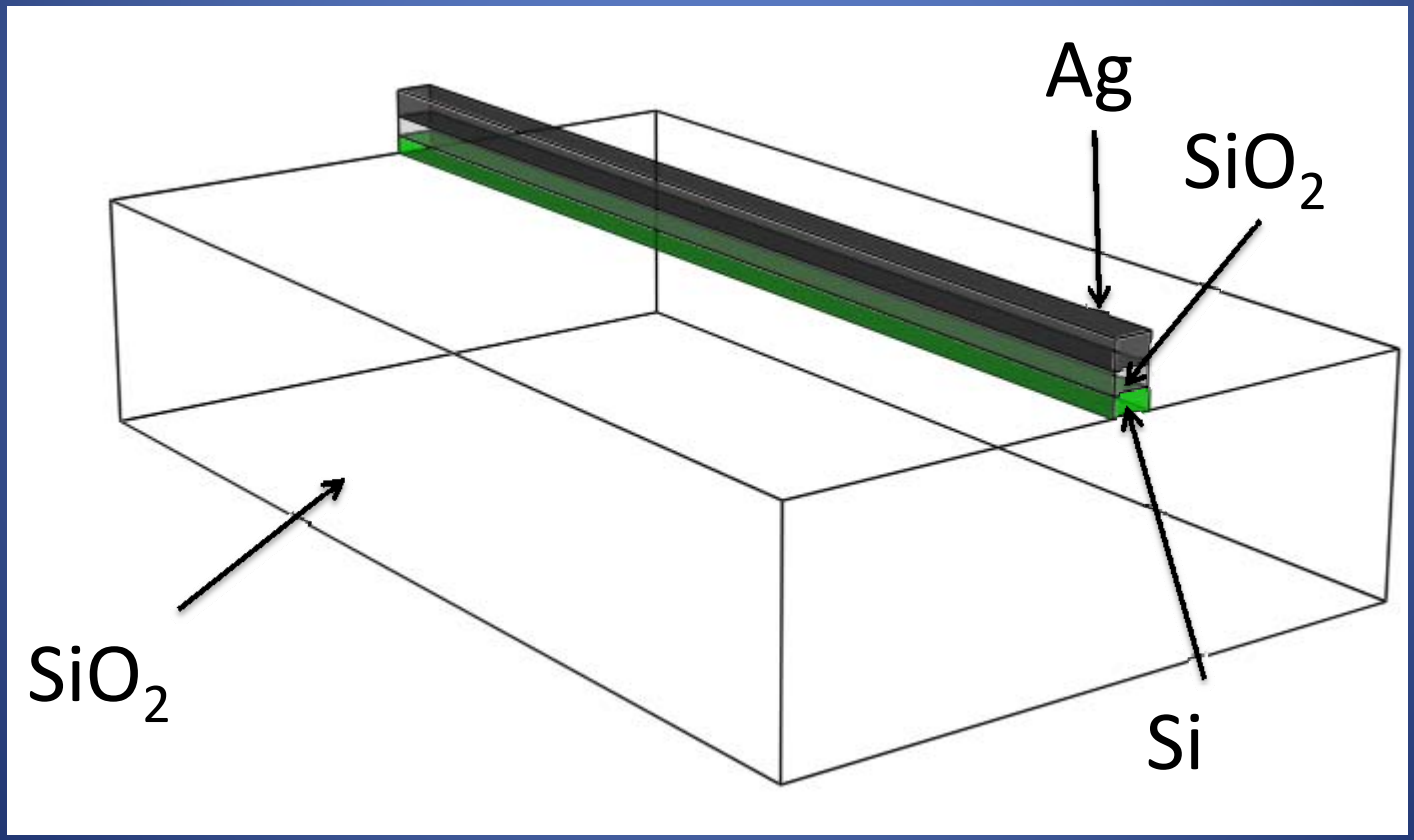
Examples (Photonics)

- The Hybrid Waveguide:

  1. Show the geometry of the waveguide
  2. Show how the EM mode is confined in the waveguide

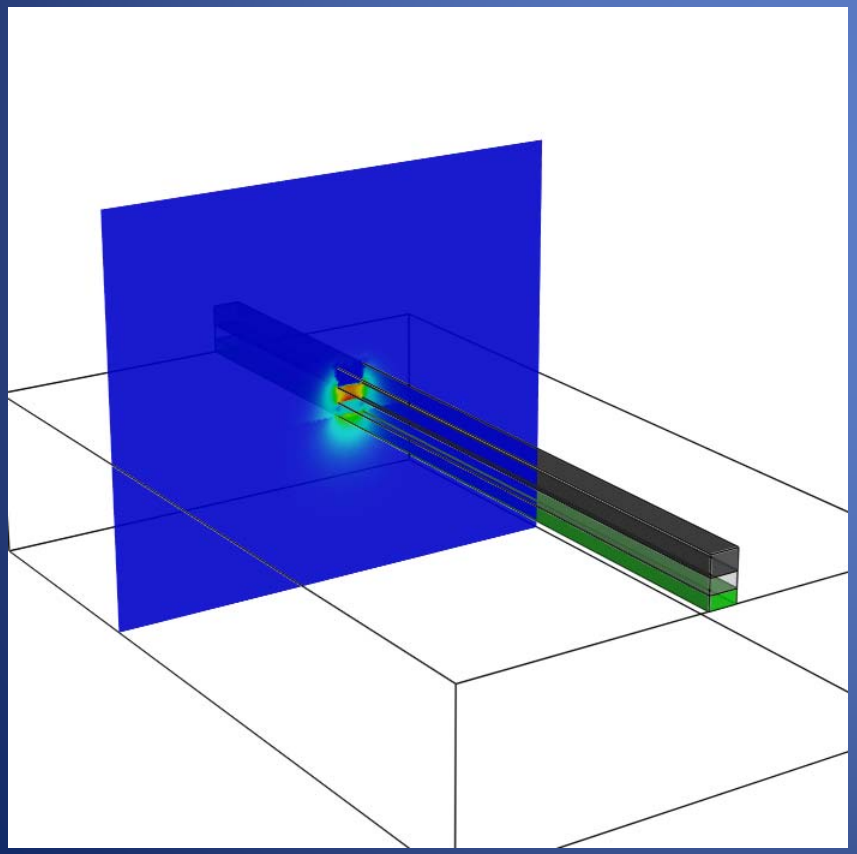
Examples (Photonics)

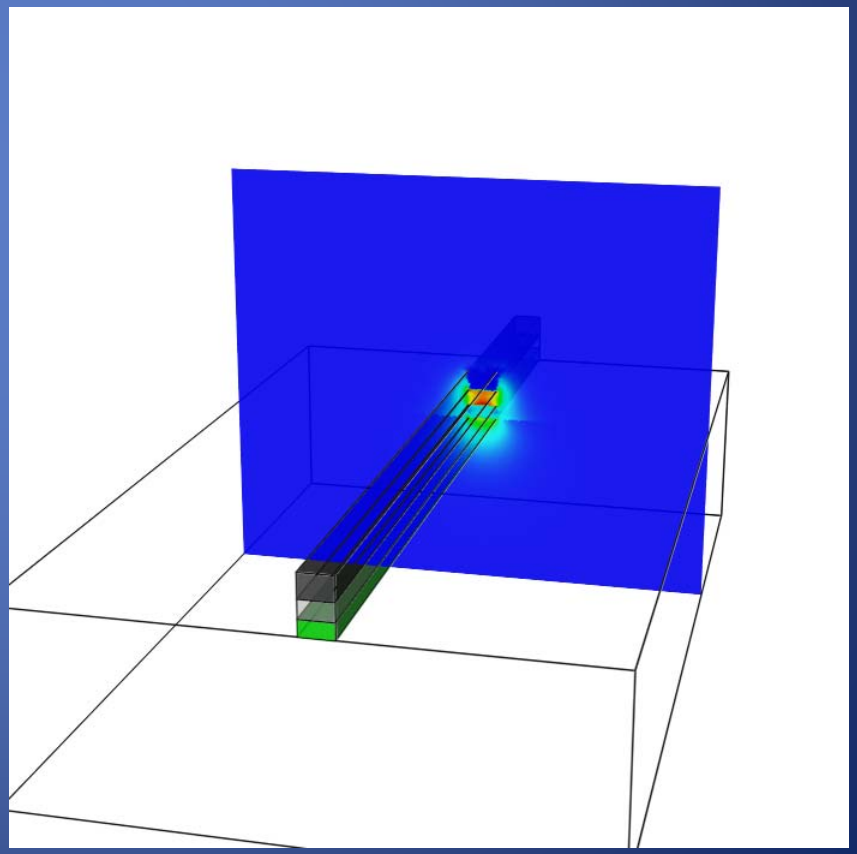
- Schematic



Examples (Photonics)

- Hybrid Mode Plot



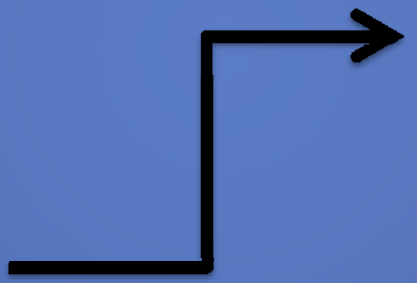


Examples (Photonics)

- Code

1. Read in Data

2. Define Scene
   - Set up camera
   - Set up Renderer and Window



```
cam1=vtk.vtkCamera()
cam1.SetFocalPoint(x0,y0,z0)
cam1.SetPosition(x1,y1,z1)
cam1.Zoom(1.5)

ren1=vtk.vtkRenderer()
ren1.SetActiveCamera(cam1)
ren1.ResetCamera()

renWin=vtk.vtkRenderWindow()
```

Examples (Photonics)

- Code Continued

3. Draw Geometry

```
Substrate=vtk.vtkCubeSource()
Substrate.SetXLength(x)
Substrate.SetYLength(y)
Substrate.SetZLength(z)
Substrate.SetCenter(cen)

mapSub=vtk.vtkPolyDataMapper()
mapSub.SetInput(Substrate.GetOutput()

aSub=vtk.vtkActor()
aSub.SetMapper(mapSub)
ren1.AddActor(aSub)
```

Examples (Photonics)

- Code Continued

4. Draw Surface Plot
   a. Setup Grid
      - Add Points and Data
   b. Set up filter and lut
   c. Map surface plot
   d. Add to scene as an actor

```
grid=vtk.vtkStructuredGrid()
grid.SetPoints(points)
grid.GetPointData().SetScalars(Efield)
```

```
filter = vtk.vtkStructuredGridGeometryFilter()
filter.SetInput(grid)
```

```
lut=vtk.vtkLookupTable()
lut.Build()
```

```
map=vtk.vtkPolyDataMapper()
map.SetInput(filter.GetOutput())
map.SetLookupTable(lut)
```

```
vecActor=vtk.vtkActor()
vecActor.SetMapper(map)
ren1.AddActor(vecActor)
```

Examples (EM)

- Ray Tracer
  - We need to understand how EM waves propagate inside very large domains like tunnels
  - Use a ray tracer to study the problem. (Written By Neeraj Sood)

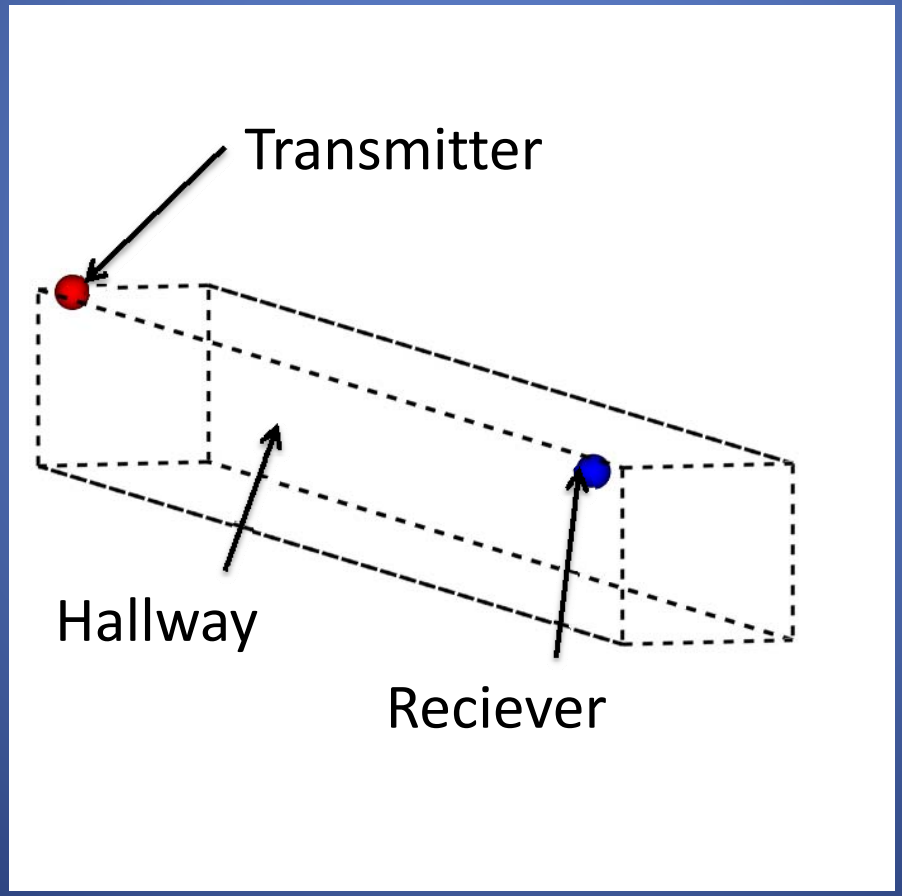
Examples (EM)

- Ray Tracer

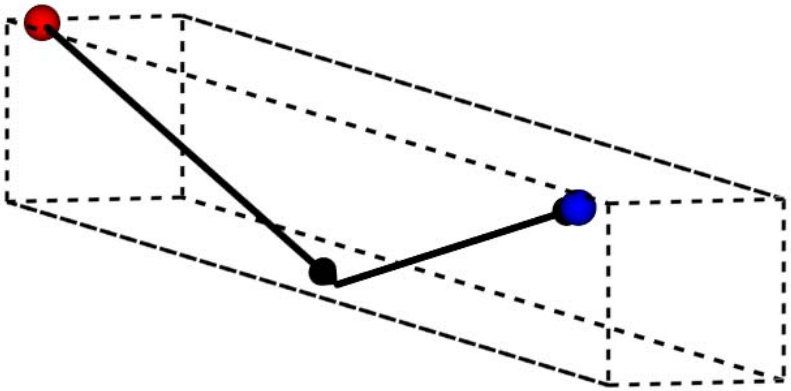
  1. Show the geometry of the problem being studied
  2. Visualize the rays in that geometry

Examples (EM)

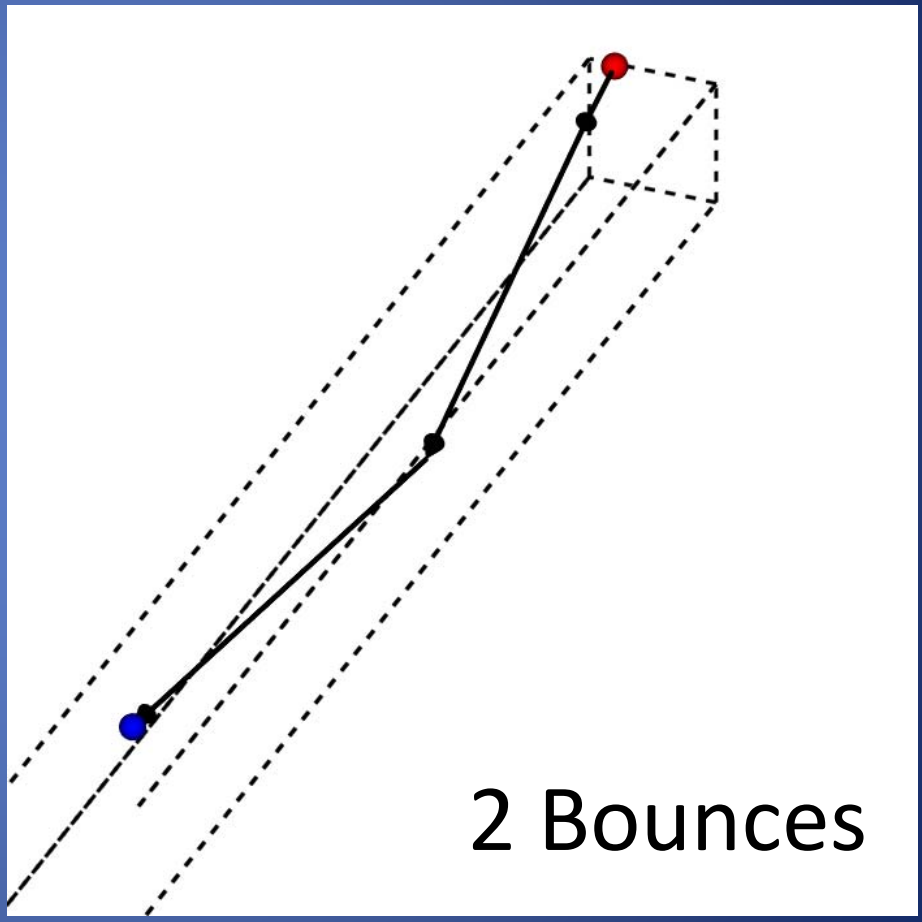
- Schematic



Examples (EM)

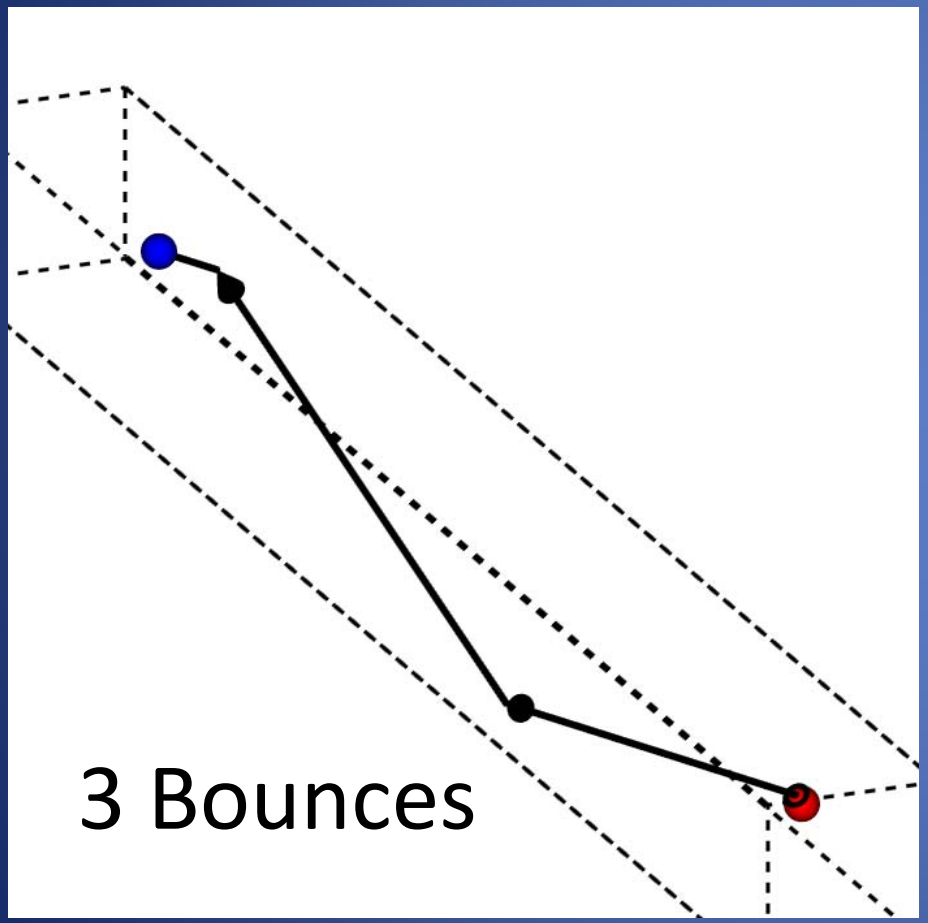


1 Bounce

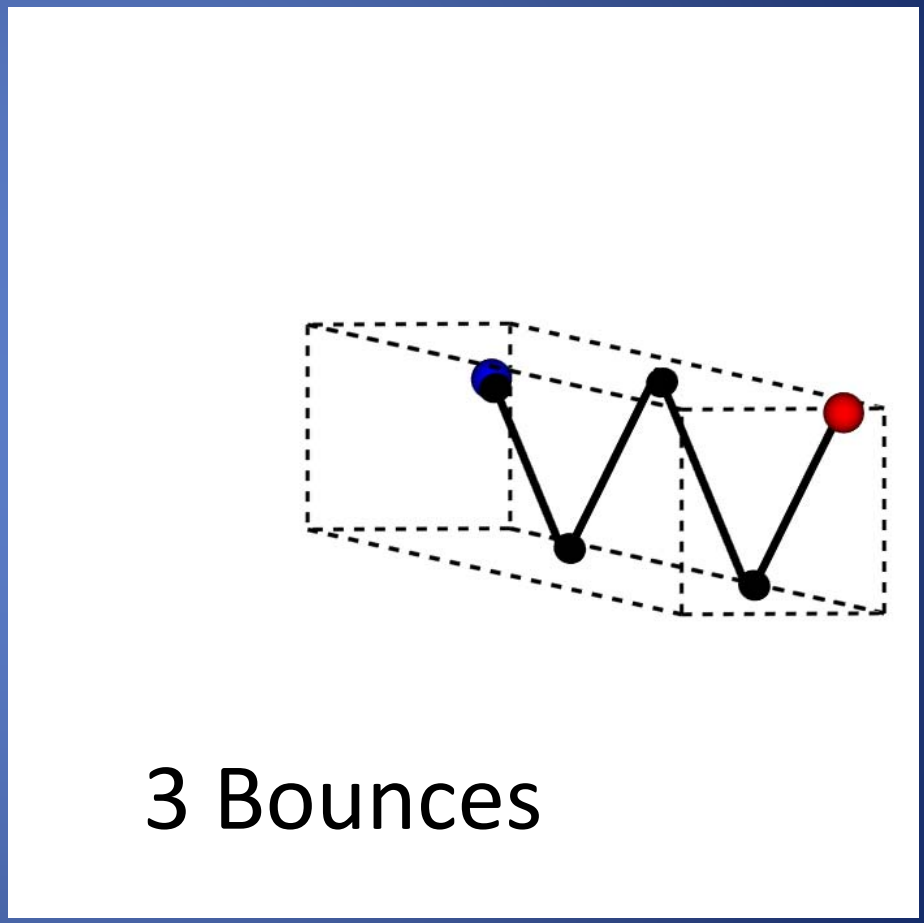


2 Bounces

Examples (EM)



3 Bounces



3 Bounces

Examples (EM)

- Code:

1. Define Scene
2. Read in Geometry
3. Draw Geometry
4. Read in Rays
5. Draw Rays
    a. Draw Arrows

VTK Pros and Cons

- Pros:
  1. A very powerful tool. Can `visualize' almost anything:
     - scalars, vectors, tensors, complex numbers..
  2. A lot of examples online
  3. Reusable code

VTK Pros and Cons

- Cons:
  1. Setup is difficult
  2. Very steep learning curve
     1. Need to learn intricacies of API
     2. Need to understand the framework
     3. Lots of implicit assumptions

Final Thoughts

- VTK is a tool for 3D visualization
- If you want to play around with 3D visualization it is worth investigating
- Code is available if you're curious