Programming with VTK ---
A High-level Visualization Toolkit

Weiguang Guan
Sharcnet
Draw a cube using OpenGL

Top face:
```
glBegin(GL_POLYGON);
    glVertex3f(-0.5, -0.5, 0.5); // P1
    glVertex3f(-0.5, 0.5, 0.5); // P2
    glVertex3f(0.5, 0.5, 0.5); // P3
    glVertex3f(0.5, -0.5, 0.5); // P4
glEnd();
```
More Primitives

- Cone
  
- Cylinder
  
- Sphere
Outline

- Overview of VTK
- Basics of VTK
- A “hello world” application
- VTK architecture
- VTK’s main classes
- More applications
- Advanced topics
- Resources
Overview of VTK

- Created in 1993
- Open Source maintained by Kitware Inc
- Powerful high-level visualization library
  - 3D graphics
  - Image processing
  - Visualization
- Cross platform (Linux, Unix, Windows, MacOS)
- 2627 C++ classes (version 6.1)
- Tcl, Java, Python wrappers
Overview of VTK (cont.)

- Shorter code/more efficient
- Many users/code contributors
- Active development
- Visualization of large-scale data
- Well-documented and resources
Basics of VTK

- vtkDataObject
  - vtkImageData
  - vtkRectilinearGrid
  - vtkStructuredGrid
  - vtkPolydata
  - vtkUnstructuredGrid
Basics of VTK (cont.)

- `vtkAlgorithm`
  - Source
    - Procedural sources
    - Reader sources
  - Filter

```
Source -> Data object
Data object -> Filter -> Data object
```
Basics of VTK (cont.)

- Connection of sources/filters
  filter1->SetInputConnection(source->GetOutputPort())

filter2->SetInputConnection(filter1->GetOutputPort())
Basics of VTK (cont.)

- Main directories of source tree
  - VTK/Common --- Core classes
  - VTK/Filtering --- Data processing in pipeline
  - VTK/Graphics --- Filters that process 3D data
  - VTK/Hybrid --- complex classes
  - VTK/Imaging --- Image processing filters
  - VTK/IO --- data file reading/writing
  - VTK/Parallel --- Parallel processing support
  - VTK/Rendering --- various rendering techniques
  - VTK/Widgets --- 3D widgets
A “hello world” app

# define source
cone = vtk.vtkConeSource()
cone.SetHeight( 3.0 )
cone.SetRadius( 1.0 )
cone.SetResolution( 10 )

# define mapper
coneMapper = vtk.vtkPolyDataMapper()
coneMapper.SetInputConnection( cone.GetOutputPort() )

#define actor
coneActor = vtk.vtkActor()
coneActor.SetMapper( coneMapper )
A “hello world” app (cont.)

# define renderer
ren = vtk.vtkRenderer()
ren.AddActor(coneActor)
ren.SetBackground(0.1, 0.2, 0.4)

# define rendering window
renWin = vtk.vtkRenderWindow()
renWin.AddRenderer(ren)
renWin.SetSize(300, 300)

# define interactor
iren = vtk.vtkRenderWindowInteractor()
iren.SetRenderWindow(renWin)
iren.Initialize()
iren.Start()
A “hello world” app (cont.)

- Add another primitive
- Set color
- Set position
- Set orientation

VTK architecture

- vtkRenderWindow
- vtkRenderWindowInteractor
- vtkRenderer
- Actor #1
- Actor #2
- Actor #3
- vtkRenderer #1
- Viewport #1
- Actor #4
- Actor #5
- vtkRenderer #2
- Viewport #2
- vtkRenderWindowInteractor
VTK architecture (cont.)

- VTK processing unit (vtkAlgorithm)
  - Time stamp of latest update
  - Time stamp of latest modification
- VTK “lazy-update” scheme
  - Up-stream Update()
  - Down-stream RequestData()
VTK’s main classes

- `vtkProp` (vtkActor, vtkVolume, vtkActor2D): position, scaling, orientation
- `vtkAbstractMapper` (vtkPolyDataMapper, vtkFixedPointVolumeRayCastMapper): rendering
- `vtkProperty`: appearance such as color, opacity, surface optical property
- `vtkCamera`: eye position, focal point, clipping planes, view frustum
- `vtkLight`: specification of lights
- `vtkRenderer`: holds props, camera, lights
- `vtkRenderWindow`: windowing
- `vtkRenderWindowInteractor`: rotating, moving, scaling
More applications

- Load and render STL data
- Load and render 3DS data
- Load and render medical image data
Advanced topics

- Event observers and callback methods
  - `AddObserver(Event, Callback)`
    - `StartEvent`
    - `EndEvent`
    - `ProgressEvent`
    - `ErrorEvent`
    - `WarningEvent`
  - Other events (picking, dragging, selecting, keyboard/mouse, or user-defined events)
Advanced topics

- I/O
  - Readers/Writers (load/save a specific dataset. e.g., vtkMultiBlockPLOT3DReader, vtkIVWriter)
  - Importers/Exporters (import/export whole scene. e.g., vtk3DSImporter, vtkIVExporter)
  - Screenshots (bmp, png, jpeg, tiff, etc)
  - Movies (vtkMPEG2Writer)
Advanced topics (cont.)

- Widgets
  - `vtkScalarBarWidget`
  - `vtkPointWidget`
  - `vtkLineWidget`
  - `vtkPlaneWidget`
  - `vtkBoxWidget`
  - `vtkImagePlaneWidget`
  - `vtkSphereWidget`
  - `vtkSplineWidget`
Advanced topics (cont.)

- Animation
  - vtkAnimationScene
  - vtkAnimationCue
Advanced topics (cont.)

- Parallel visualization
  - Multi-threading
  - MPI-based Parallelism
Resources

- Official web [www.vtk.org](http://www.vtk.org)
- Wiki [http://www.vtk.org/Wiki/VTK](http://www.vtk.org/Wiki/VTK)
- Mailing lists
  - [vtkusers@vtk.org](mailto:vtkusers@vtk.org)
  - [vtk-developers@vtk.org](mailto:vtk-developers@vtk.org)
- Examples
Resources

- Textbook “Visualization Toolkit: An Object-Oriented Approach to 3D Graphics”
- “VTK User’s Guide”
Questions?