Visualization

Introduction

Joseph A. Insley

August 11, 2014

Argonne Leadership Computing Facility
Here’s the plan...

- Examples of visualizations
- Visualization resources
- Data and transformations
- Visualization tools and formats
- Data representations
- Production vis
Arterial Blood Flow

Data courtesy of: George Karniadakis and Leopold Grinberg, Brown University
Climate

Data courtesy of: Mark Taylor, Sandia National Laboratory; Rob Jacob, Argonne National Laboratory; Warren Washington, National Center for Atmospheric Research
Aerospace (Jet Nozzle Noise)

Data courtesy of: Anurag Gupta and Umesh Paliath, General Electric Global Research
Cosmology

Data courtesy of: Salman Habib, Katrin Heitmann, Argonne National Laboratory
Astrophysics

Data Analysis: Normalized Density Difference

Data courtesy of: Michael Norman, Robert Harkness, Rick Wagner, San Diego Supercomputer Center
High Performance Visualization Resource: Tukey

- **96 AMD Dual Opteron 6128 Compute Nodes**
  - 16 total CPU cores per node
  - 64 GB RAM
- **2 NVIDIA Tesla M2070 GPUs, 6 GB RAM each**
- **6 terabytes of CPU RAM, 1.1 terabytes of GPU RAM**
- **Peak GPU Performance: Over 98 TeraFLOPS**
- **Cross-Mounted Filesystem with Mira**

![Image of Tukey resource]
Visualization Algorithms (Transformations)

- **Structure**
  - **Geometric**
    - Translate, rotate, scale coordinates
    - Topology remains unchanged
  - **Attribute**
    - Transform or create new data attributes
  - **Combined**

- **Type**
  - **Scalar** (single value)
  - **Vector** (array of values)
  - **Tensor** (matrix of values)
  - **Combined**
Data Domain Decomposition: Regular Grid

- Regularly sized/spaced grid of cells, each holds a single value (per variable)
- Data domain is divided among available processes
- Additional “ghost” cells are required to ensure accuracy
Data Domain Decomposition: Regular Grid

- Regularly sized/spaced grid of cells, each holds a single value (per variable)
- Data domain is divided among available processes
- Additional “ghost” cells are required to ensure accuracy
Data Domain Decomposition: Adaptive Mesh Refinement (AMR)

- Puts increased detail in regions where things are changing more rapidly.
- Can increase computational performance.
- Results in smaller data sets.
Data Domain Decomposition

Adaptive Mesh Refinement (AMR)

- Puts increased detail in regions where things are changing more rapidly.
- Results in smaller data sets
- Enables greater detail
Data Domain Decomposition: Particle-based

- Keep track of individual particles
- Decomposition could be based on particles, or spatial extents
- Can project them onto a grid
  - combine (e.g. average) all particles in each grid cell
In situ analysis and data reduction

- Incorporate analysis routines into the simulation code
  - operate on data while it is still in memory

- Potential for significant reduction the I/O demands
  - application scientist identifies features of interest
  - compress data of less interest

All Sorts of Tools

- Visualization Applications
  - VisIt
  - ParaView
  - EnSight
- Domain Specific
  - VMD, PyMol, RasMol
- APIs
  - VTK: visualization
  - ITK: segmentation & registration
- GPU performance
  - vl3: shader-based volume rendering
  - Scout: GPGPU acceleration
- Analysis Environments
  - Matlab
  - Parallel R
- Utilities
  - GnuPlot
  - ImageMagick
- Visualization Workflow
  - VisTrails
ParaView & VisIt vs. vtk

- **ParaView & VisIt**
  - General purpose visualization applications
  - GUI-based
  - Scriptable
  - Extendable
  - Built on top of vtk (largely)

- **vtk**
  - Programming environment / API
  - Additional capabilities, finer control
  - Smaller memory footprint
  - Requires more expertise (build custom applications)
Data File Formats (ParaView & VisIt)

- VTK
- Parallel (partitioned) VTK
- VTK MultiBlock (MultiGroup, Hierarchical, Hierarchical Box)
- Legacy VTK
- Parallel (partitioned) legacy VTK
- EnSight files
- EnSight Master Server
- Exodus
- BYU
- XDMF
- PLOT2D
- PLOT3D
- SpyPlot CTH
- HDF5 raw image data
- DEM
- VRML
- PLY
- Polygonal Protein Data Bank
- XMol Molecule
- Stereo Lithography
- Gaussian Cube
- Raw (binary)
- AVS
- Meta Image
- Facet
- PNG
- SAF
- LS-Dyna
- Nek5000
- OVERFLOW
- paraDIS
- PATTRAN
- PFLOTRAN
- Pixie
- PuReMD
- S3D
- SAS
- Tetrade
- UNIC
- VASP
- ZeusMP
- ANALYZE
- BOV
- GMV
- Tecplot
- Vis5D
- Xmdv
- XSF
Data Wrangling

- **XDMF**
  - XML wrapper around HDF5 data
  - Can define
    - data sets
    - subsets
    - hyperslabs

- **vtk**
  - Could add to your simulation code
  - Can write small utilities to convert data
    - Use your own read routines
    - Write vtk data structures
  - C++ and Python bindings
Data Organization

Format
- Existing tools support many flavors
- Use one of these formats
- Use (or write) a format converter
- Write a custom reader for existing tool
- Write your own custom vis tool

Serial vs. Parallel/Partitioned
- Single big file vs. many small files: middle ground generally best
  - vtk data types
  - XDMF for VisIt and ParaView
  - Custom
Data Organization

- **Serial vs. Parallel/Partitioned**
  - Performance trade-offs
    - vtk/paraview: serial files all data read on head node, partitioned and distributed
    - vtk/paraview: parallel files: serial files partitioned across processes, read in parallel

Performance example:

- **Single serial .vtu file (unstructured grid)**
  - Data size: ~3.8GB
  - Read time on 64 processes: > 15 minutes
    - most of this was spent partitioning and distributing

- **Partitioned .pvtu file (unstructured grid)**
  - Data size: ~8.7GB (64 partitions)
  - Read time on 64 processes: < 1 second
Visual Cues

**Position**
Where in space the data is

**Length**
How long the shapes are

**Angle**
Rotation between vectors

**Direction**
Slope of a vector in space

**Shapes**
Symbols as categories

**Area**
How much 2-D space

**Volume**
How much 3-D space

**Color Saturation**
Intensity of a color hue

**Color Hue**
Usually referred to as color
Data Representations: Volume Rendering
Data Representations: Volume Rendering

- Turn 2- and 3-dimensional datasets into 2D images
- Approximation: Volume ray casting
Data Representations: Volume Rendering

- Turn 2- and 3-dimensional datasets into 2D images
- Approximation: Volume ray casting
- Parallelize: domain decomposition
Compositing

- Each process renders its chunk
- Chunks get reassembled into final image
Data Representations: Glyphs

- 2D or 3D geometric object to represent point data
- Location dictated by coordinate
  - 3D location on mesh
  - 2D position in table/graph
- Attributes graphical entity dictated by attributes of a data
Data Representations: Glyphs

- VisIt & ParaView:
  - good at this

- vtk:
  - same, but again requires more effort

- gnuplot:
  - good at 2D plots, tables of numbers
Data Representations: Contours (Isosurfaces)

- A Line (2D) or Surface (3D), representing a constant value
- VisIt & ParaView:
  - good at this
- vtk:
  - same, but again requires more effort
Data Representations: Cutting Planes

- Slice a plane through the data
  - Can apply additional visualization methods to resulting plane
- VisIt & ParaView & vtk good at this
- VMD has similar capabilities for some data formats
Data Representations: Streamlines

- From vector field on a mesh (needs connectivity)
  - Show the direction an element will travel in at any point in time.
- VisIt & ParaView & vtk good at this
Molecular Dynamics Visualization

- **VMD:**
  - Lots of domain-specific representations
  - Many different file formats
  - Animation
  - Scriptable
  - Not parallel

- **VisIt & ParaView:**
  - Limited support for these types of representations

- **VTK:**
  - Anything’s possible if you try hard enough
gnuplot

- Graphing utility
- Command-line driven
Annotation, compositing, scaling...

- ImageMagick
  - convert, composite, montage, etc.
Annotation, compositing, scaling...

- ImageMagick
  - scale, fade
Movie Creation

- VisIt and ParaView can spit out a movie file (.avi, etc.)
  - can also spit out individual images

- Combine multiple segments of frames
  - Create a directory of symbolic links to all frames in order

- ffmpeg: Movie encoding
  - ffmpeg -sameq -i frame.%04d.png movie.mp4
More info...

- www.alcf.anl.gov/user-guides/tukey
- www.visitusers.org/
- www.paraview.org/wiki/ParaView
- www.imagemagick.org/