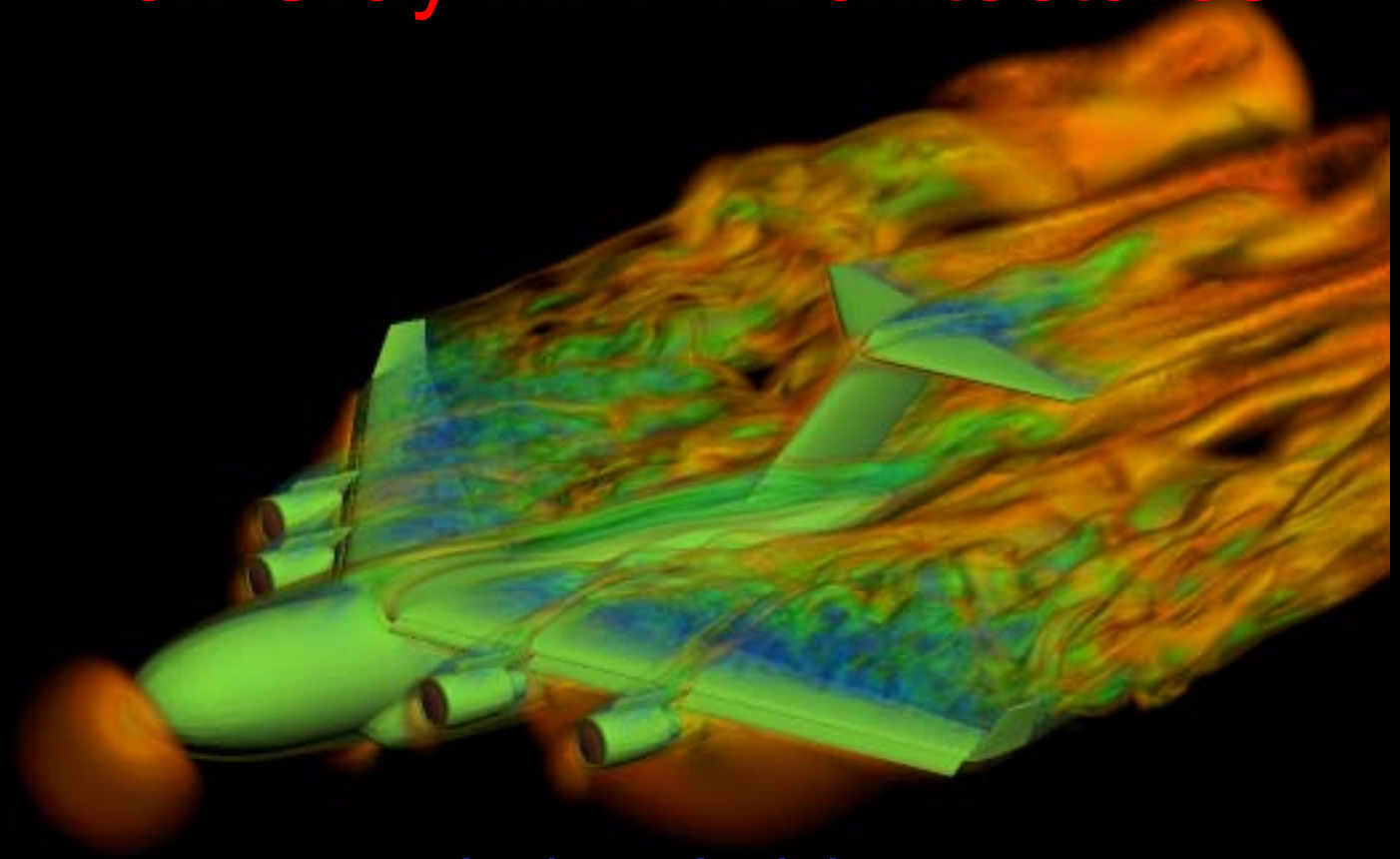
Large Scale Scientific Visualization on Cray MPP Architectures



Andrew A. Johnson
Army HPC Research Center / NetworkCS, Inc.
Minneapolis, Minnesota

AHPCRC

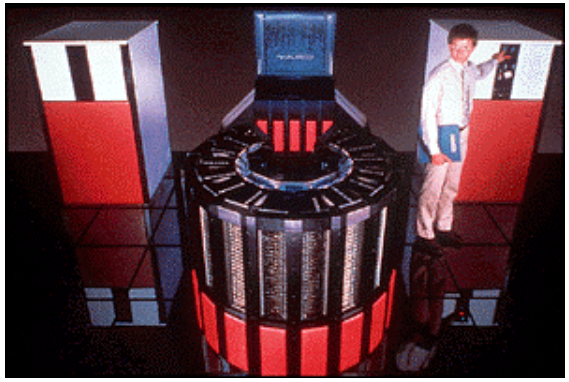
AHPCRC Background

- Program is in its 13th year
- Funded by the U.S. Army through the Army Research Laboratory
- Funding for hardware acquisition is through the Department of Defense High Performance Computing Modernization Program

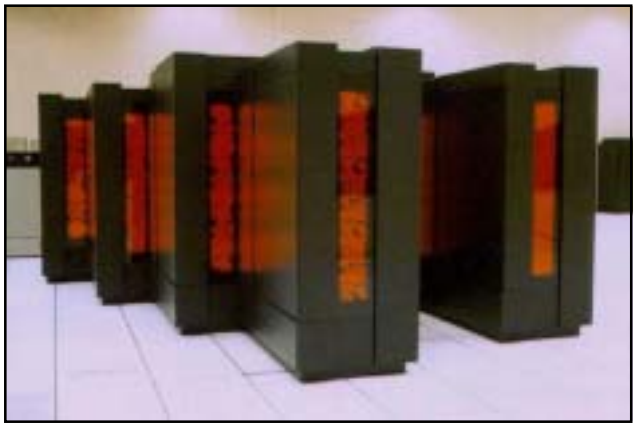
- Government-University-Industry partnership for research and development of HPC applications and systems
  - University of Minnesota
  - Clark Atlanta University, Jackson State University, Howard University, Florida A&M University, University of North Dakota
  - Network Computing Services, Inc.

AHPCRC

AHPCRC Computing Resources



1990 Cray Vector Systems



1991 Thinking Machines CM-5
896 Processors (Serial #1)

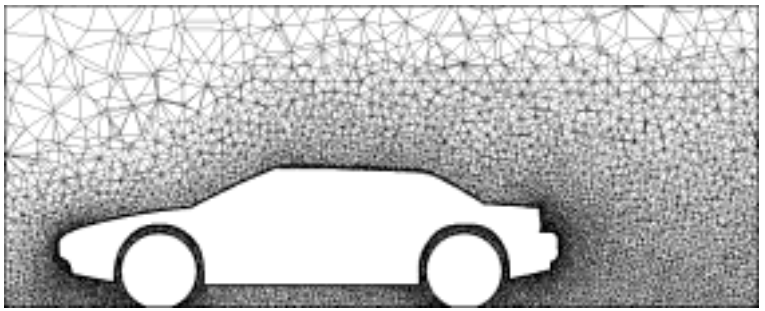
1998 Cray T3E-1200
1088 Processors (Serial #1)

2002-2003 Cray X1
64 Processors LC
128 Processors (Mid 2003)

AHPCRC

These days, scientific visualization isn't really about graphics. Its more about data management. Graphics is the easy part.

- Data storage and fast access
  - Data locality
- Parallel computing
  - Scalability
- Network transfer protocols
- Heterogeneous computing
- Feature Extraction
- Compression / Un-compression
  - Geometry
  - Images / Animations
- Interface design and usability
  - Portability



AHPCRC

Overview

- Problems & Bottlenecks found in Data Visualization
- Client-Server Framework
- Parallel Implementation
- “Presto” Overview
- Advanced Rendering Capabilities
- Examples / Large Data Sets

Mesh-based scientific data sets found in CFD, CSM, and other disciplines. 3D time-dependent results involving scalor and vector variables.

AHPCRC

Large-Scale Data Visualization

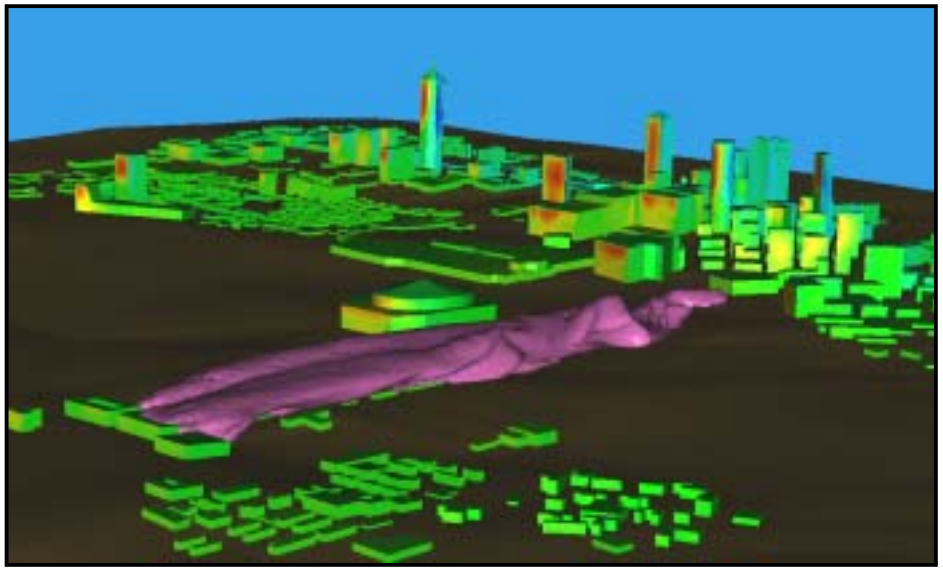
- It is becoming very difficult to visualize 3D simulation data sets with “traditional” methods
  - Download entire data set on to your local workstation
  - Load the entire data set into system memory
  - Interactively process and visualize using a workstation with 1 or 2 CPUs
- Today’s HPC power allows much larger data-sets to be solved
  - Computed and residing on a remote HPC parallel system
  - Applications can scale up to 1 billion unstructured elements
  - Gbytes in total simulation data set size
- Should use the same HPC system that computed/generated the data set to visualize it also

AHPCRC

Recent Case Study

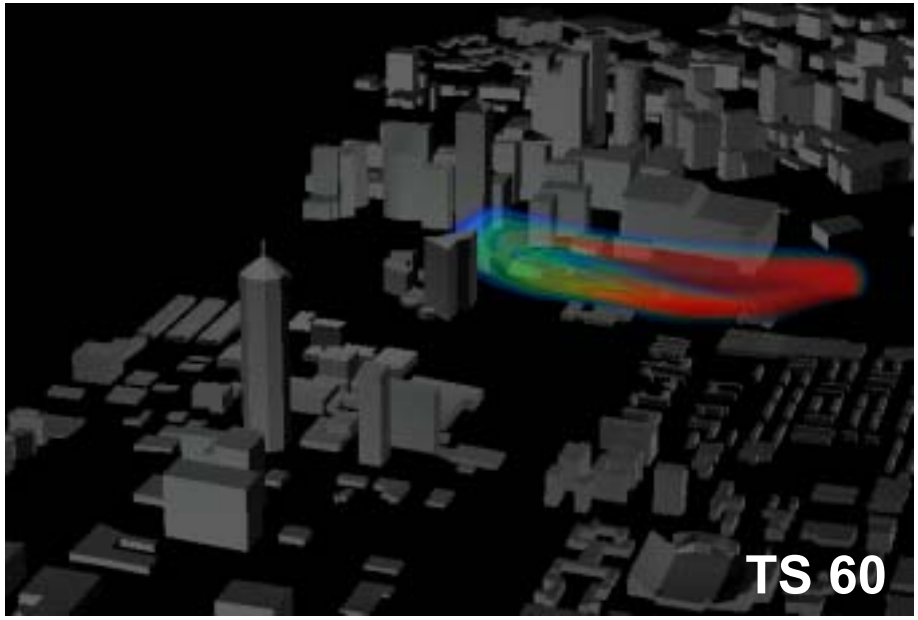
Contaminant Dispersion in Atlanta
CAU Researchers (Atlanta, Georgia)

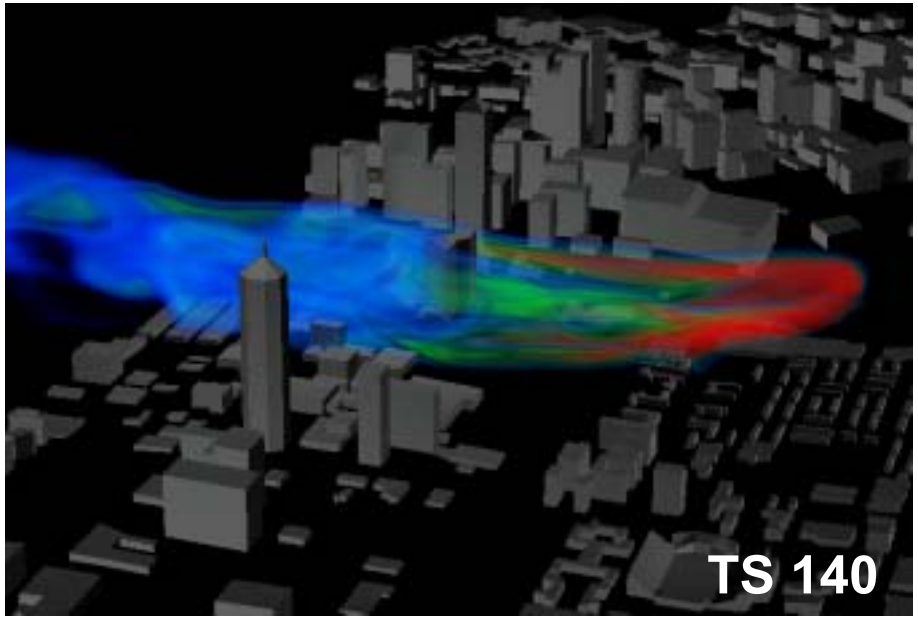
- Geometric Model
  - 76K Control Points
  - 10K Model Faces
- 3D Unstructured Mesh
  - 55 Million Tetrahedrons
  - 10 Million Nodes
- 856 Time Steps
  - Data file per time step
  - 6 Variables per node
- 512 T3E Processors Used
  - Minneapolis, Minnesota

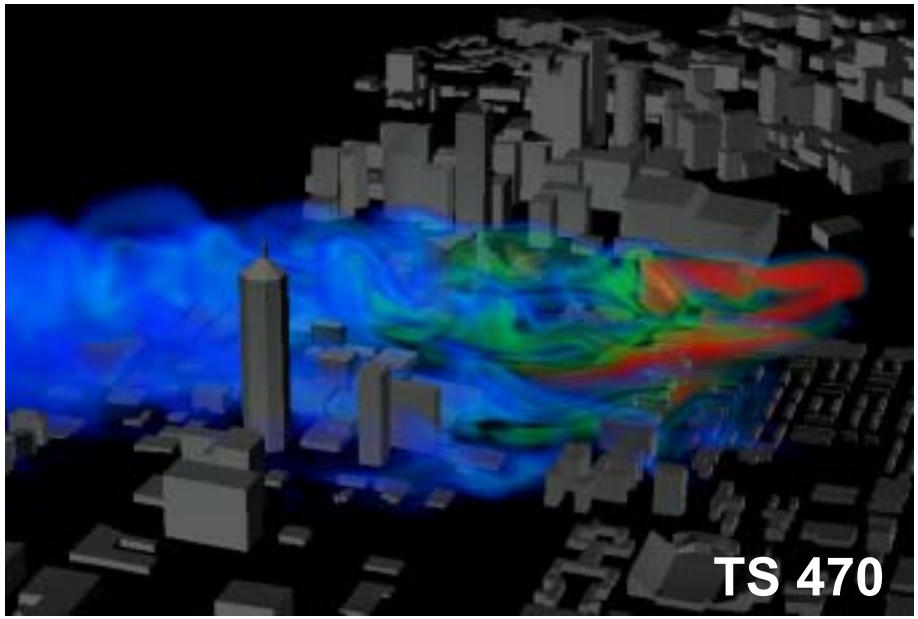


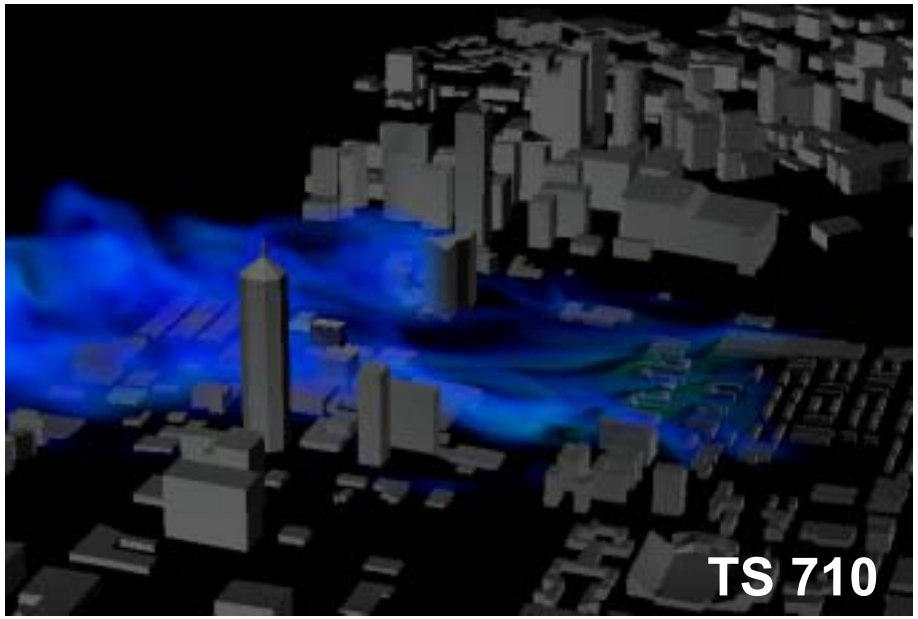
- **2 GBytes to store the mesh**
- **0.45 GBytes per time step**
- **385 GBytes for all data files**

AHPCRC

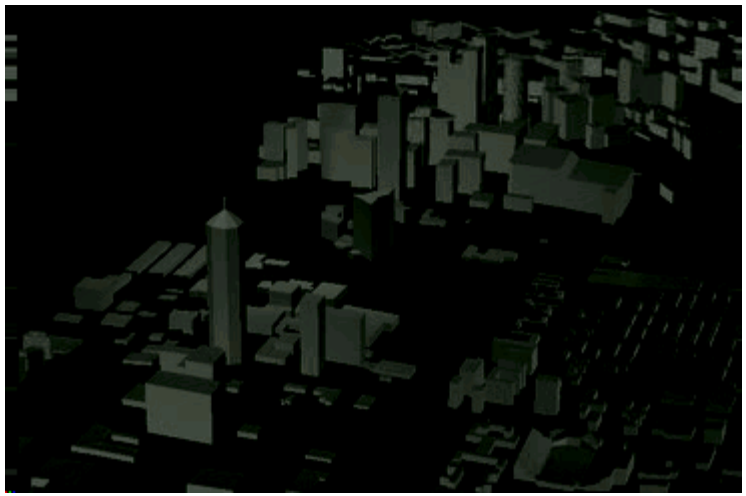




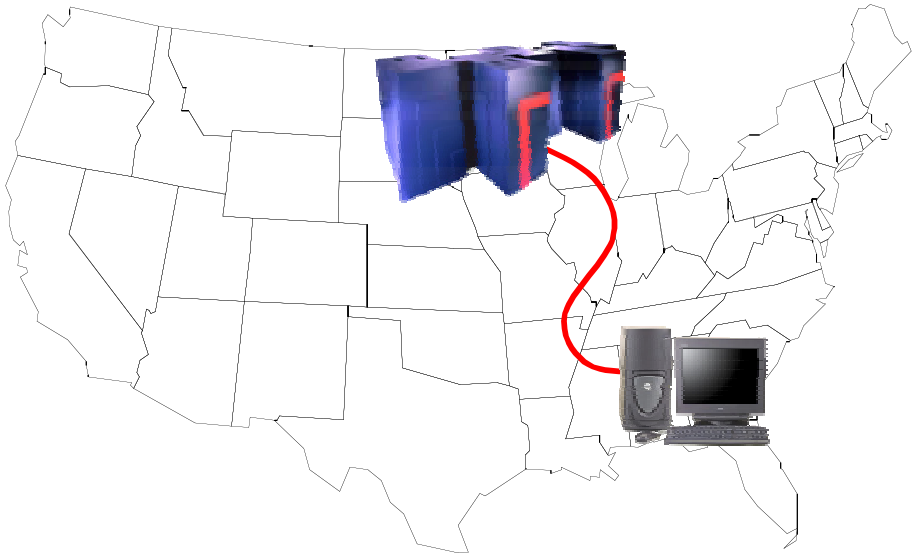




AHPCRC



Data Transfer Times



- Want to post-process and visualize the results
- Dedicated T-1 Line (Minneapolis to Atlanta)
- Mesh: 2.96 Hours at maximum rate (1.5 MBits per sec)
- Single Data File: 40 Minutes
- All Data Files: 23.7 Days
- Need to find a place to store 387 GBytes
- Several other simulations will be (are being) carried out
  - Different conditions and contaminant release locations

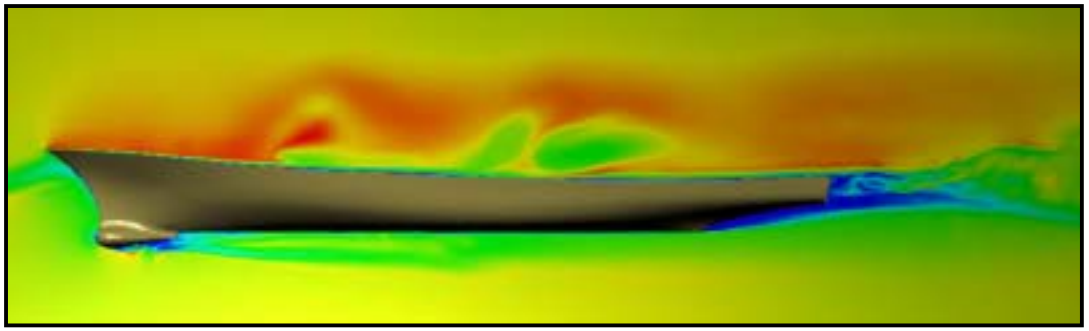
AHPCRC

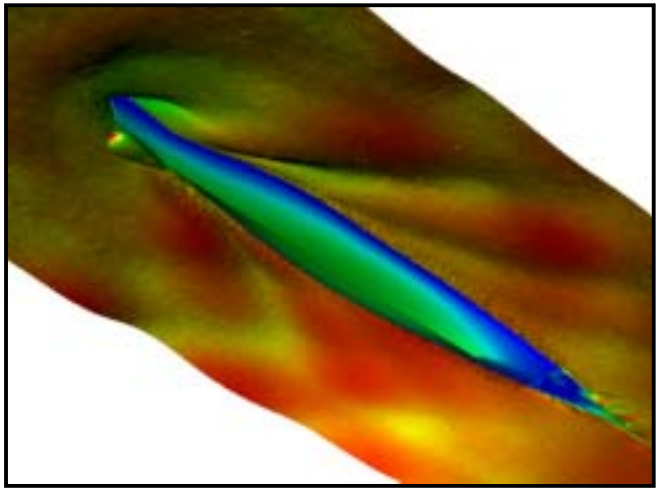
What Does This Mean?

- Today's new HPC systems allow simulations to be performed at extremely large scales
- Simulation results should not (can not) move from the HPC site where it was created
  - Store the results on the large HPC work disks
- Anything that "touches" the data set needs to be parallel (probably MPI-based) on a HPC platform
  - Pre-processing
  - Numerical Simulation
  - Post-processing and analysis
- Almost all numerical simulation tools should be built to support large remote data sets and run in parallel

AHPCRC

Large Scale Results





- **1 Billion** unstructured (tetrahedral) elements
  - 1 to 30 million are “normal”
  - Implicit (GMRES-Based) solver
- Roughly 170 million nodes and 850 million equations
- 1056 Processors of the Cray T3E were used
  - Largest we could fit into memory (512 MB per processor)
- Creating the mesh and visualization was a challenge
- Roughly 115 GFlops sustained performance
- Visualized “interactively” with Presto

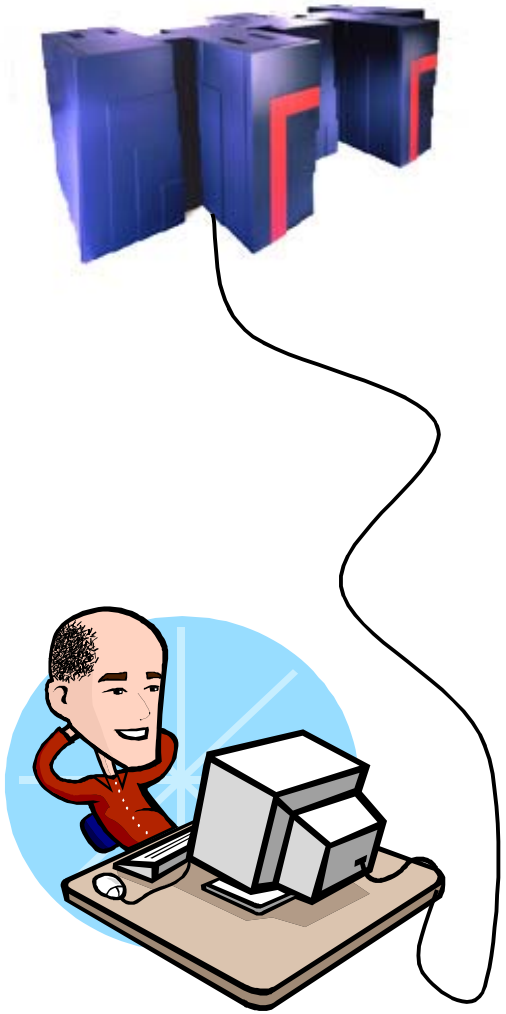
AHPCRC

Problems & Bottlenecks

- Data Locality
  - Large data sets are computed and reside on remote HPC systems
  - 1 to 400+ GBytes becoming common
  - Transferring across the Internet and storing locally is impractical
- Processing Power
  - Significant computational and system memory required
  - HPC parallel resources needed
- Expense
  - Users typically have basic desktop systems (PC, Linux)
  - Limited to no access to an expensive visualization system or laboratory
- Ease-of-use

Remote Computing

- Control and use remote HPC resources from a user's desktop environment
  - HPC resources located at a central site
- Fast, Efficient, Effective, and Transparent
- Full control of simulation process
  - Geometric Modeling
  - Automatic Mesh Generation
  - Solver Set-up, Control, and Monitoring
  - Visualization of Computed Data Sets



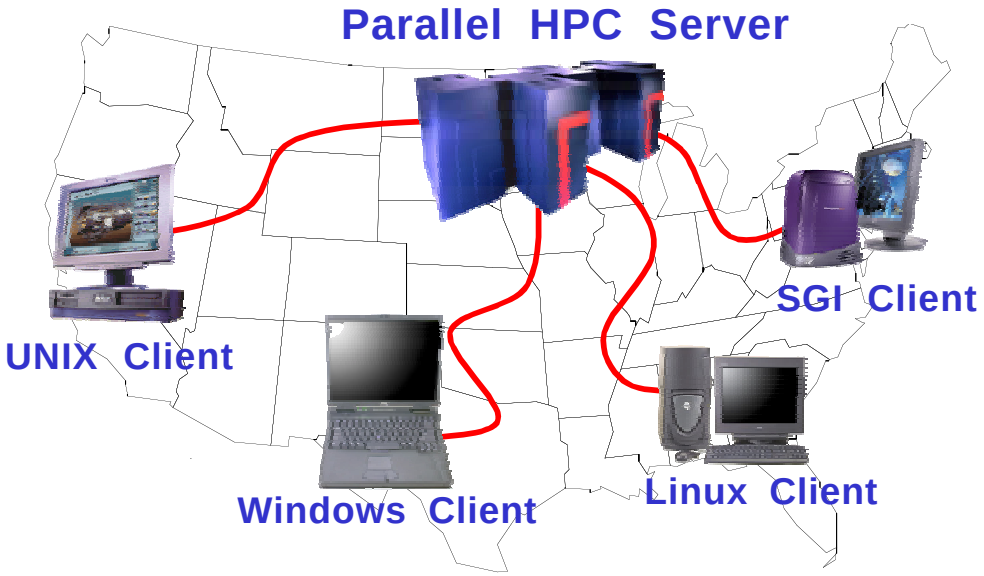
- Large-Scale data visualization is the greatest challenge
  - If it works for visualization, all other steps in the simulation process are easier

AHPCRC

Presto Visualizer

Parallel Scientific Visualization of Remote Data Sets

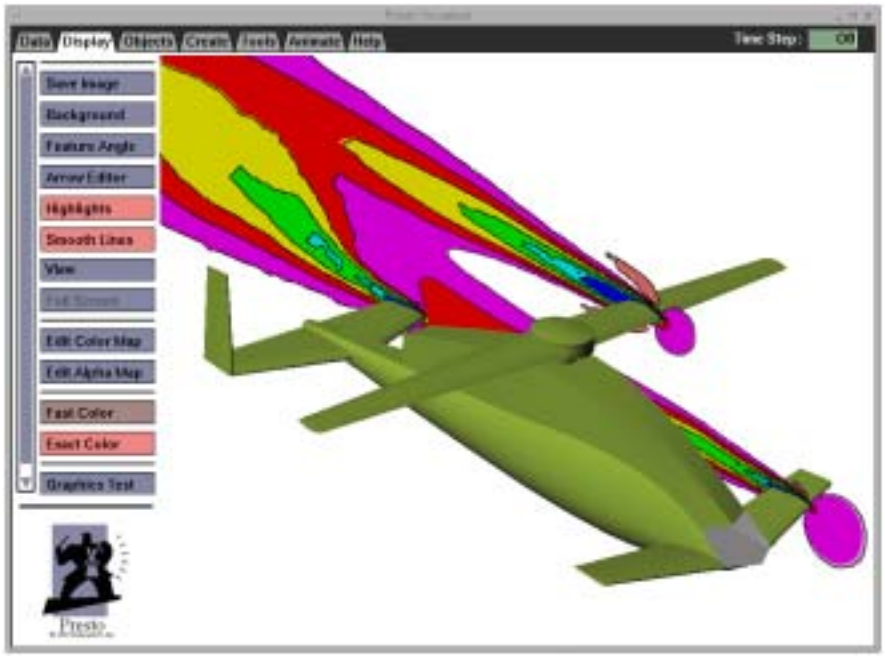
- Client-Server Model
- Remote Parallel Server
  - Distributed memory (MPI)
  - Unstructured Grids
    - Any type or mixed elements
  - Runs where the data was generated
  - Handles all data manipulations and generates all geometry (polygons)
- Local Desktop Client
  - Handles user interface and all 3D visualization (polygon surfaces)
  - Portable (UNIX, Windows)
  - Low memory
  - Only dependent upon the workstation's OpenGL performance

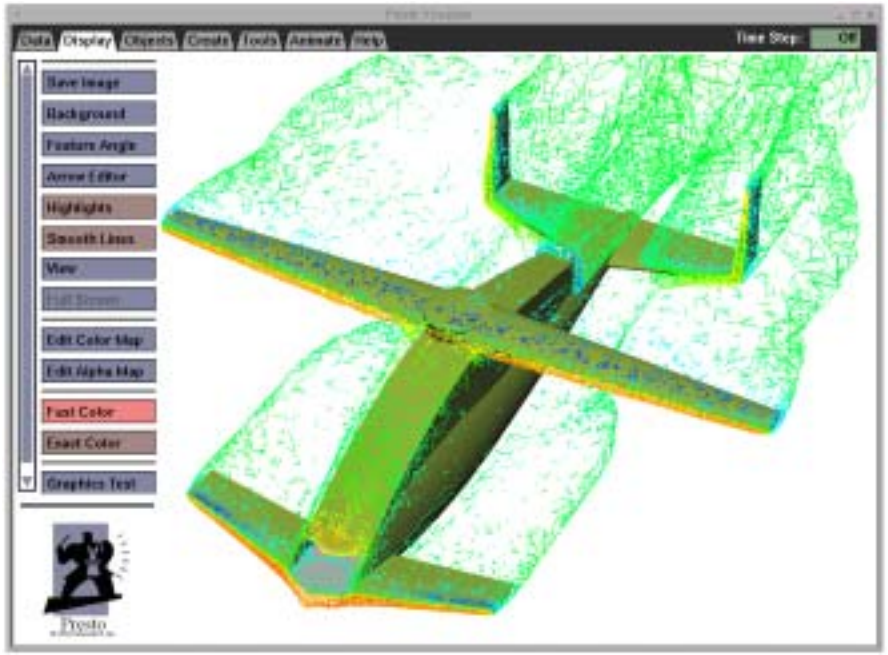


AHPCRC

Capabilities

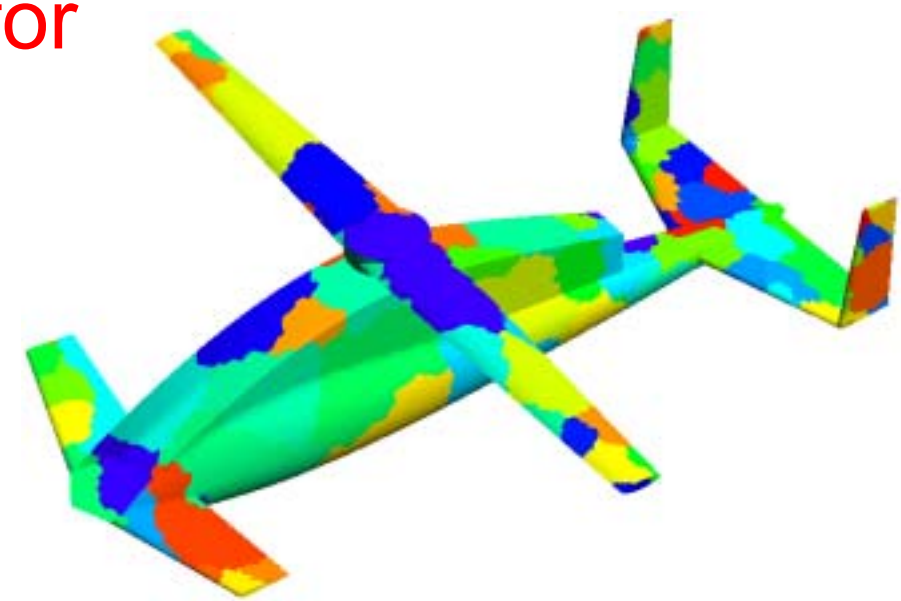
- Visualization accomplished through…
  - Boundary shadings
  - Iso-surfaces
  - Cross-sections
  - Streamlines
  - Volume rendering
- Time dependent data sets and animation capabilities
- Size of the data set is fully scalable using more processors
- Used over long distances and a variety of network capabilities





AHPCRC

Parallel Implementation for Large Data Sets

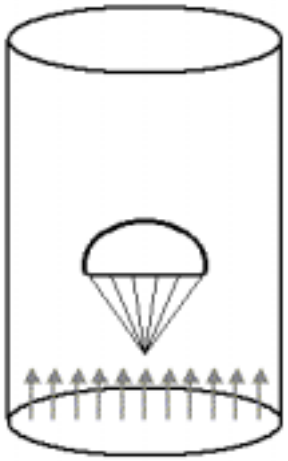


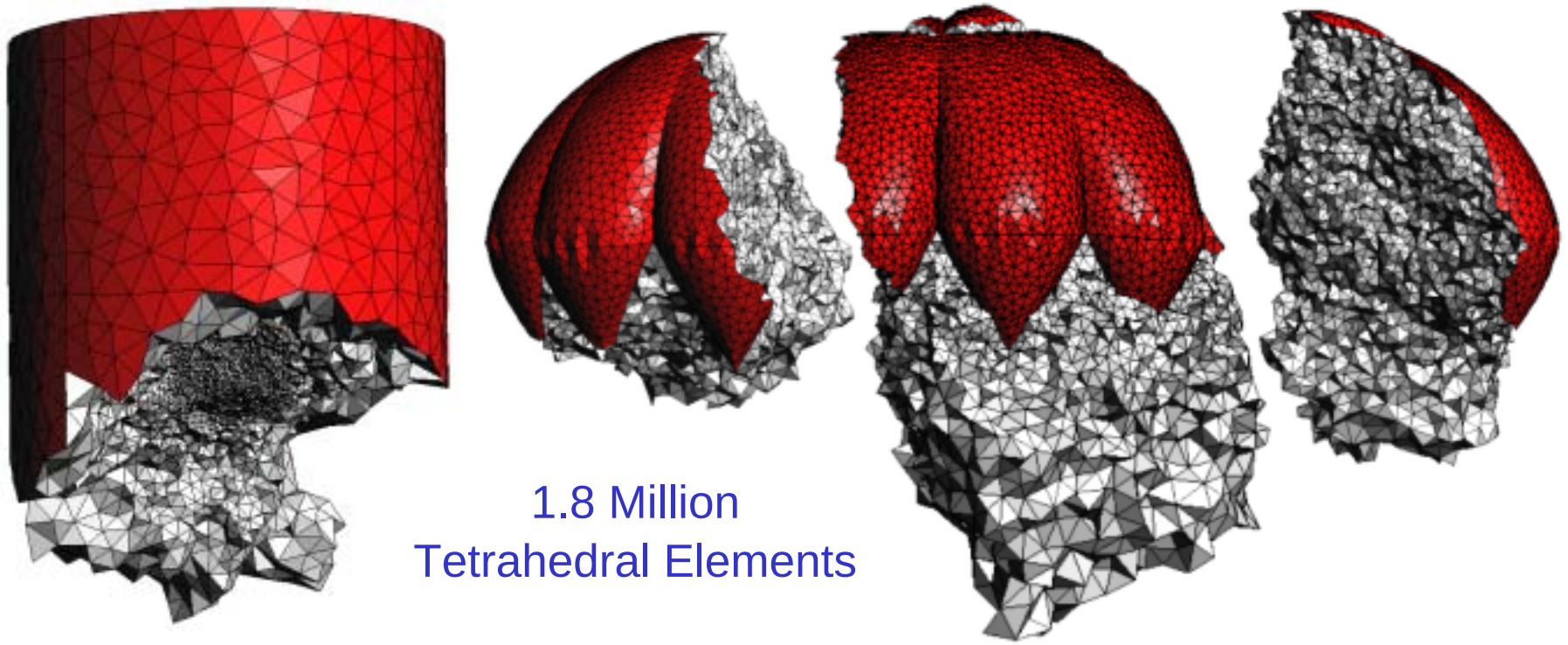
- Almost all HPC systems are parallel
  - Distributed memory
  - Same system that generated the data set
  - T3E, IBM-SP, SGI-MP, PC Clusters
- Common and portable parallel library such as MPI
- Effectively utilizes HPC resources for large data sets
- Fully scalable
  - Computational and memory

AHPCRC

Mesh / Workload Distribution

- Mesh partitioning and parallel re-distribution
  - In-house RCB algorithms or ParMETIS
  - Same number of elements and nodes in each partition
- Communication and data structures for inter-processor connectivity

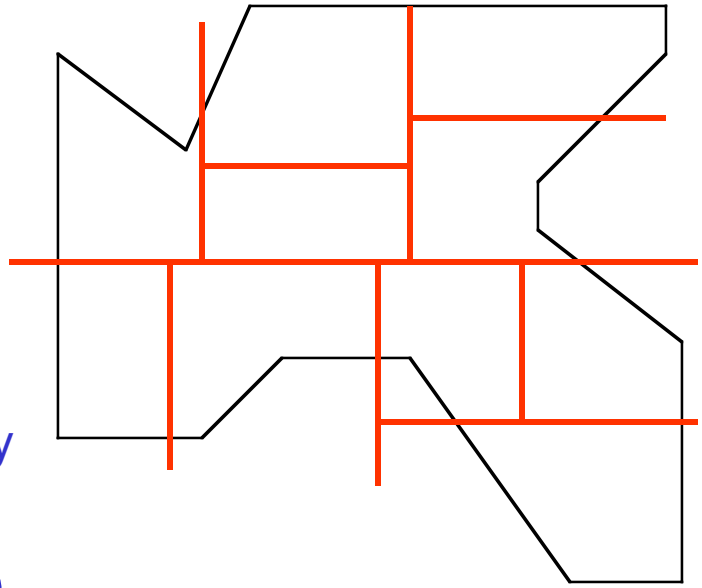




1.8 Million
Tetrahedral Elements

AHPCRC

Partitioning the Elements

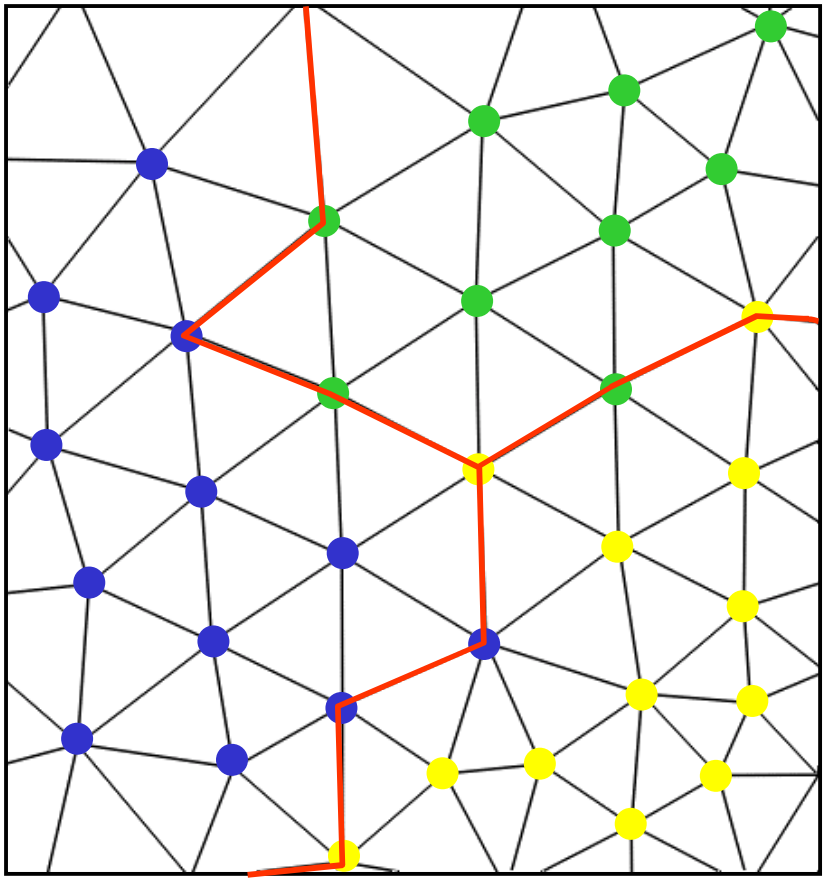


- METIS - AHPCRC/UofM CSci Group
  - Publicly/freely available library
  - Parallel implementation (ParMETIS)
  - High quality partitions (low edge cuts)
  - Fast but can (in some cases) use large memory
  - Partitions any given graph (The Dual)
- Recursive Center Bisection (RCB Partitioning)
  - Recursively sub-divides the mesh along X, Y, or Z planes
  - Partitions a list of (X,Y,Z) coordinates
    - Coordinate of element centers (not “easy” to get)
  - Dual not needed
  - Parallel implementation
  - Fast and low memory
  - Less-optimal partitions
    - Example (2% comm cost with METIS, 2.3% comm cost with RCB; T3E)

Assign Nodes to Processors

- A node referenced by elements all on the same processor will be assigned to that processor
- Nodes referenced by elements on different processors will be assigned at “random” to one of the referring processors

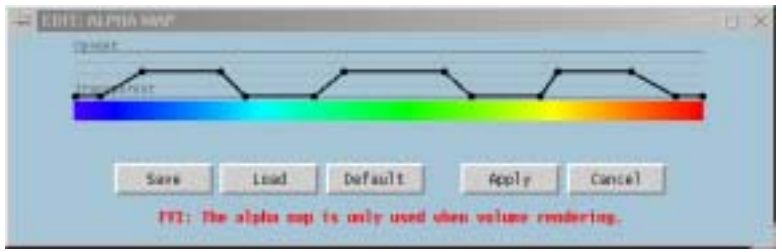
- A local (on-processor) set of nodes (copies) will be created on each mesh partition

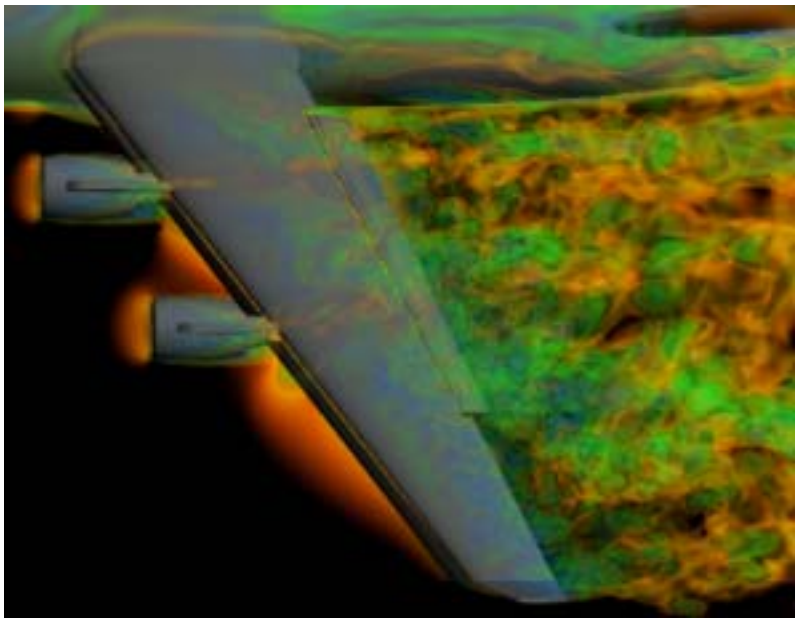


AHPCRC

Advanced Rendering Capabilities

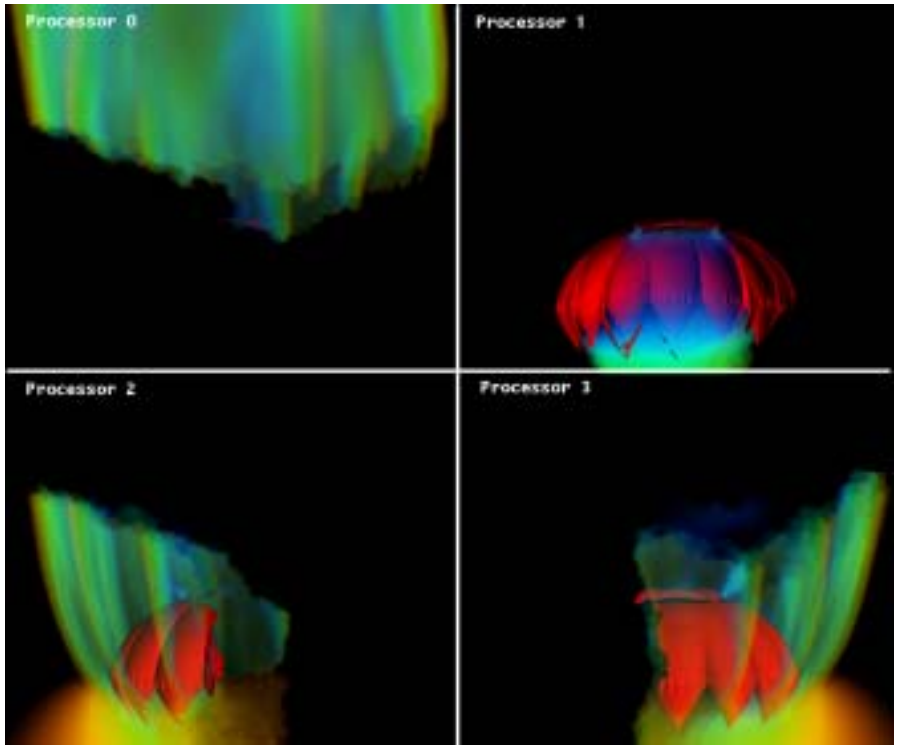
- Parallel rendering engine built within the server (polygons)
  - Large data set visualization
  - Download the final image to client for display
- Parallel volume rendering
  - Unstructured meshes
  - Fully parallel
  - Transparency and color-map used
  - “Exact” algorithm based on ray tracing
  - Detailed images of the entire volume

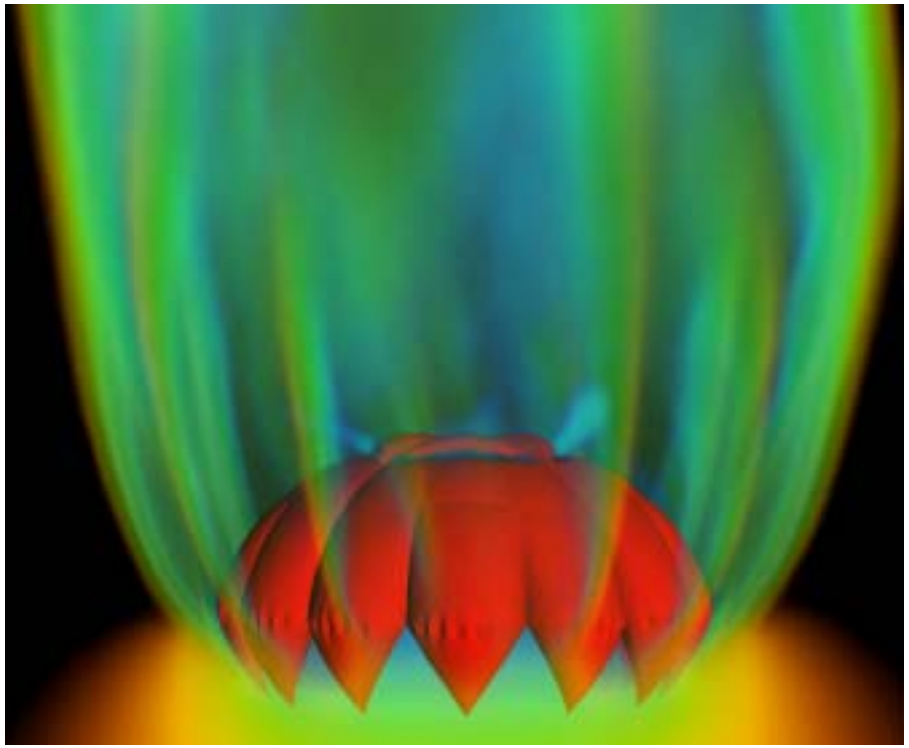




Parallel Algorithm

- Volume rendering of each mesh partition
- Parallel compositing algorithm
- Download the final image to client for display
- Newer algorithms are currently being tried

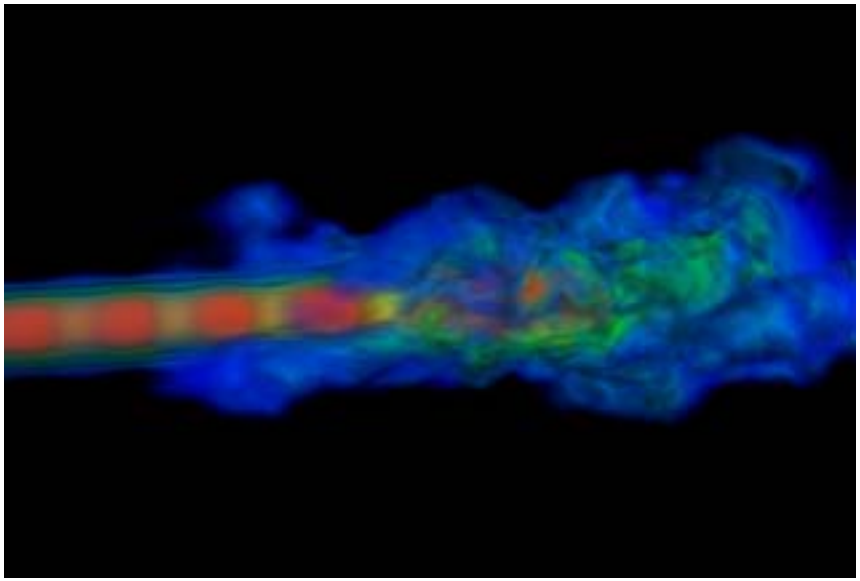


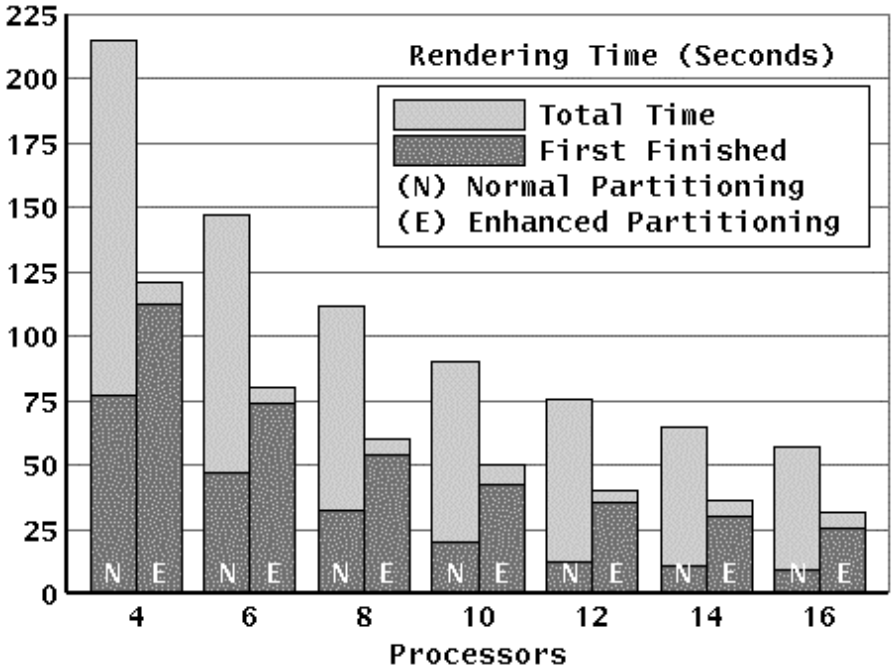


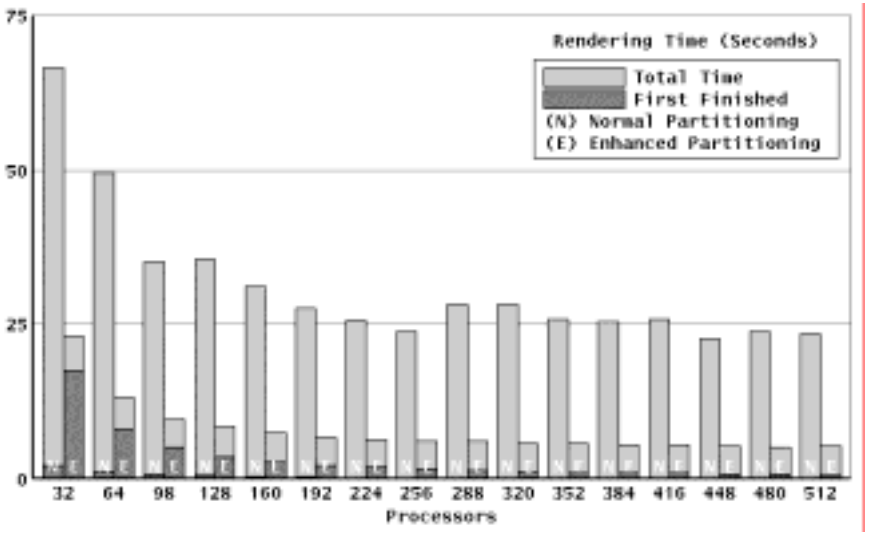
AHPCRC

Preliminary Performance Analysis

- Mesh re-partitioning algorithm used for better load-balancing during volume rendering



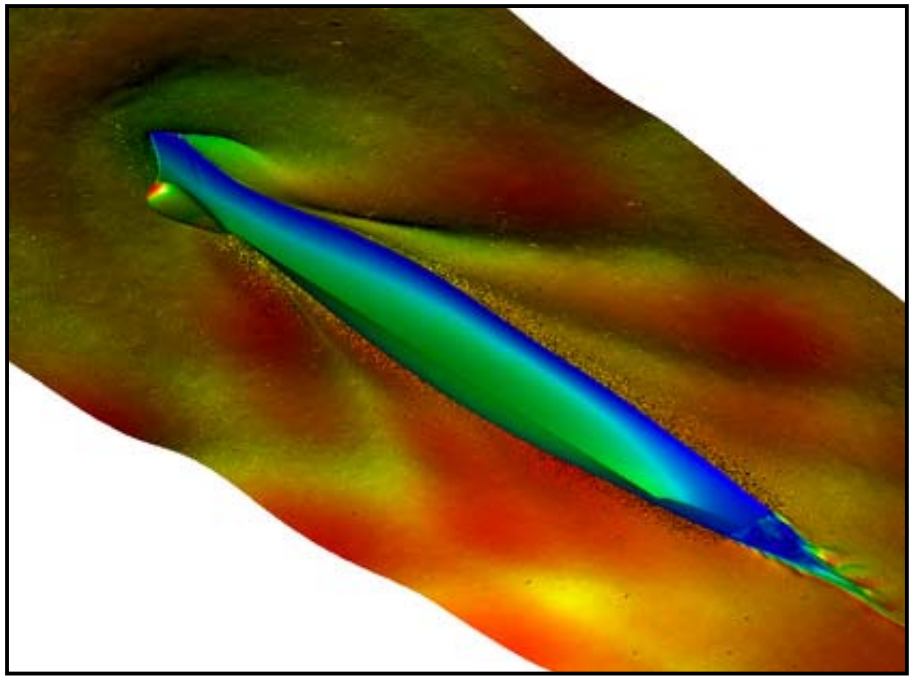


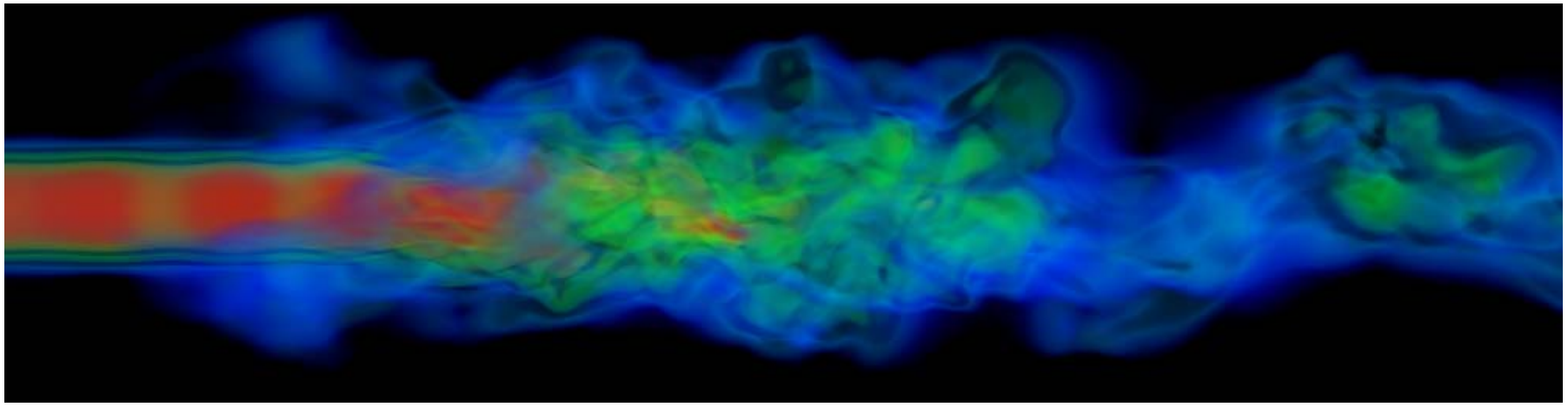


AHPCRC

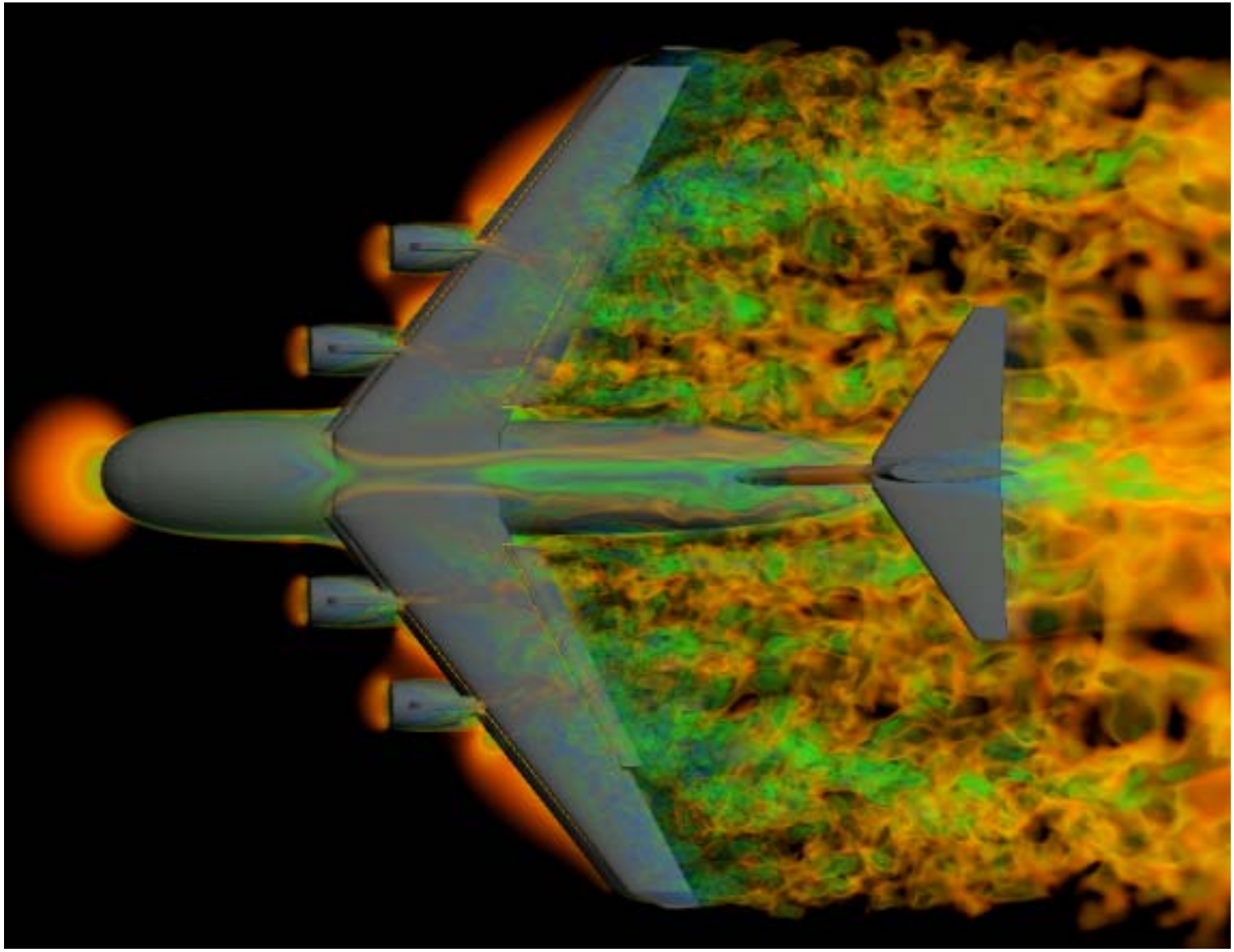
Large-Scale Visualization

- Direct Parallel Rendering
  - Directly render an image in parallel
    Download the final image for display
- Computational Hydrodynamics
  - Prediction of ship wake characteristics
  - 1 Billion tetrahedral elements
- Computational Fluid Dynamics
  - Unstable high-speed jet of air
  - 1.25 Billion tetrahedral elements
  - Direct parallel volume rendering





AHPCRC



Airflow past a cargo aircraft in a take-off configuration. Mesh contains 243 million tetrahedral elements and 41 million nodes (160 million equations). Shown is a volume-rendered image of velocity magnitude.

AHPCRC

Current Development

- "Production" Version 2.0
- Increased security and automation
  - Kerberos user authentication and encryption
  - Automated server start-up
- Native structured-mesh support
- More data formats and readers
  - Support for Solid Mechanics (EPIC) data sets
- Additional features
  - Geometry display (wire-frame, feature angle, …)
  - Particle traces and animated streamlines
  - Interactive volume rendering

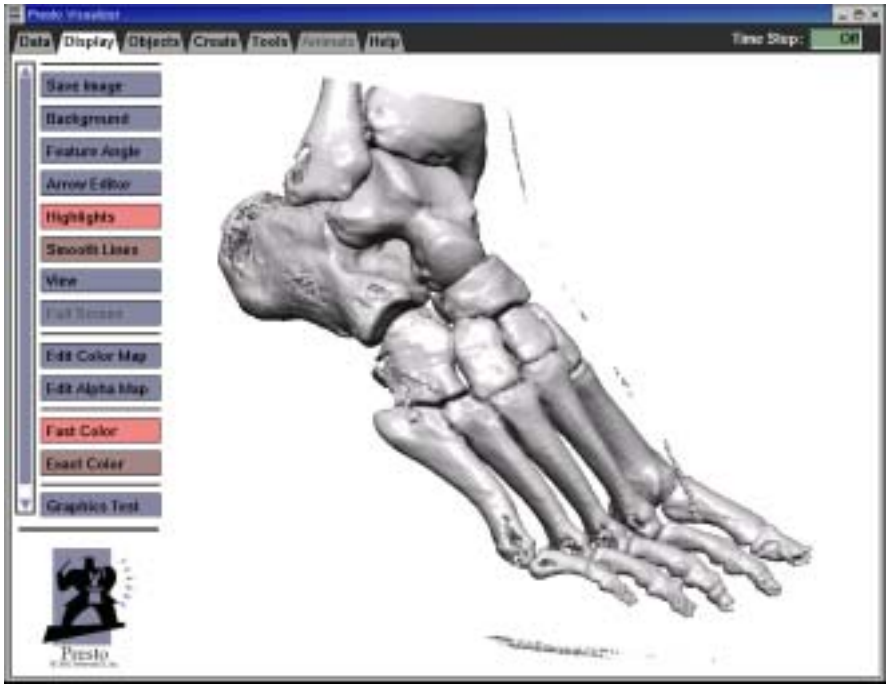
- Real-time updates (visualize results on-the-fly)
- Built-in numerical simulation capabilities
- Modular implementation for expandability

AHPCRC

Structured-Mesh Support

- Structured mesh solvers are vary common
- Support data sets with “implied” element connectivity
- Special algorithms required for geometry extraction
  - Implied element connectivity
  - Lower memory requirements
  - Faster algorithms
- Simplified algorithms for volume rendering

- Weather simulations
  - MM5
- CFD
  - Finite difference schemes
- Others
  - Medical (CAT,MRI)



AHPCRC