In-Situ Visualization for Direct Numerical Simulation of Turbulent Combustion

Hongfeng Yu

Sandia National Laboratories, Livermore, CA

Joint work with

Chaoli Wang (MTU), Ray Grout (Sandia),

Jackie Chen (Sandia), Kwan-Liu Ma (UCD)

Background

- Scientific Simulations
  - Increasing amount of data
  - Efficient and effective solutions
- Data Analysis and Visualization
  - Post-processing
  - Co-processing
- I/O and Network Bandwidth Bound
  - Data reduction

In-Situ Visualization

- Transform and Reduce Data During Simulations

- Related Work
  - Globus (1992), Parker (1995), Tu (2006) ...

- Challenges
  - Integration
  - Workload balancing and scalability
  - Low cost

In-Situ Visualization for S3D Combustion Simulations

- S3D Time Advance Loop

Integrate Field equations | Advance tracer Particles | Save restart files

  - A simulation for lifted flame stabilized
    - 1.3×10^9 grid points, 22 species
    - 140GB restart file / timestep, output every 200 timestep : interesting effects may occur more rapidly than this!
      - May not be recovered in post-processing
      - Significant I/O overhead in post-processing

In-Situ Visualization for S3D Combustion Simulations

- Incorporate In-Situ Analysis

Integrate Field equations	Advance tracer Particles	Perform In-Situ Analysis	Save restart files

  - Rendering
  - Feature extraction and tracking
  - Data reduction

  ……

In-Situ Visualization for S3D Combustion Simulations

- Incorporate In-Situ Analysis

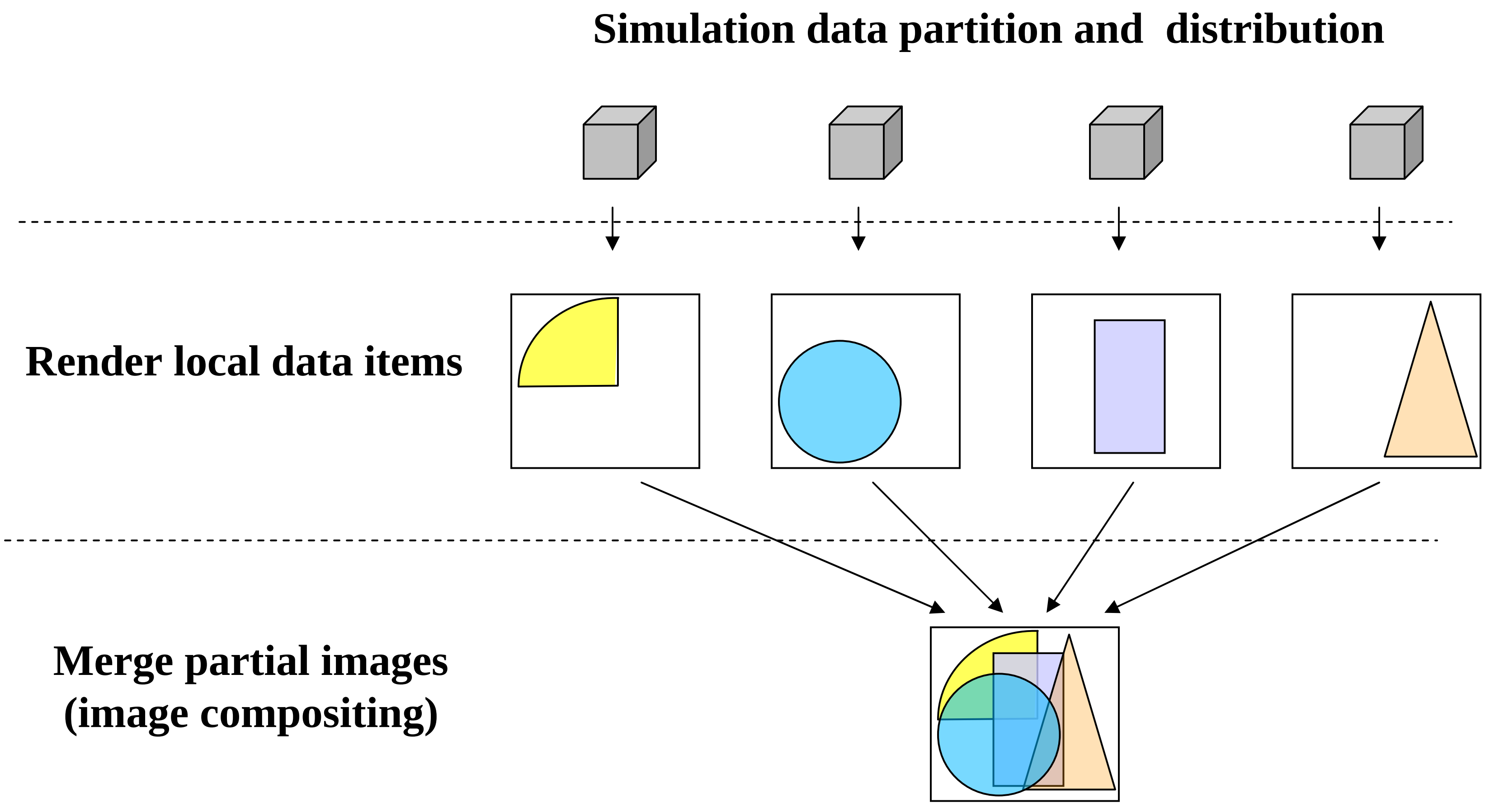
Integrate Field equations → Advance tracer Particles → Perform In-Situ Analysis → Save restart files

  - **Rendering: parallel volume and particle rendering**
  - Feature extraction and tracking
  - Data reduction

  ……

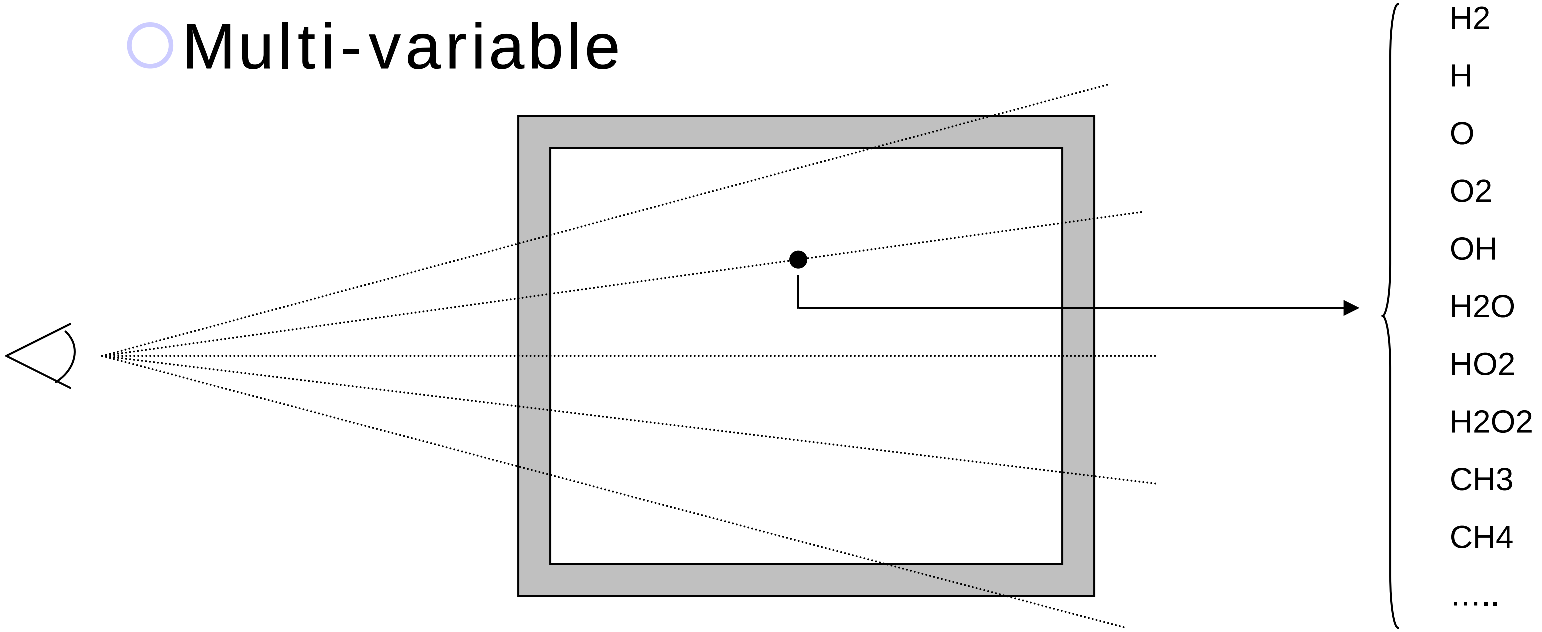
Parallel Rendering

- Sort-last Parallel Rendering



Parallel Rendering

- Volume Rendering
  - Boundary data exchange
    - Diagonal communication elimination
  - Ray casting
  - Multi-variable

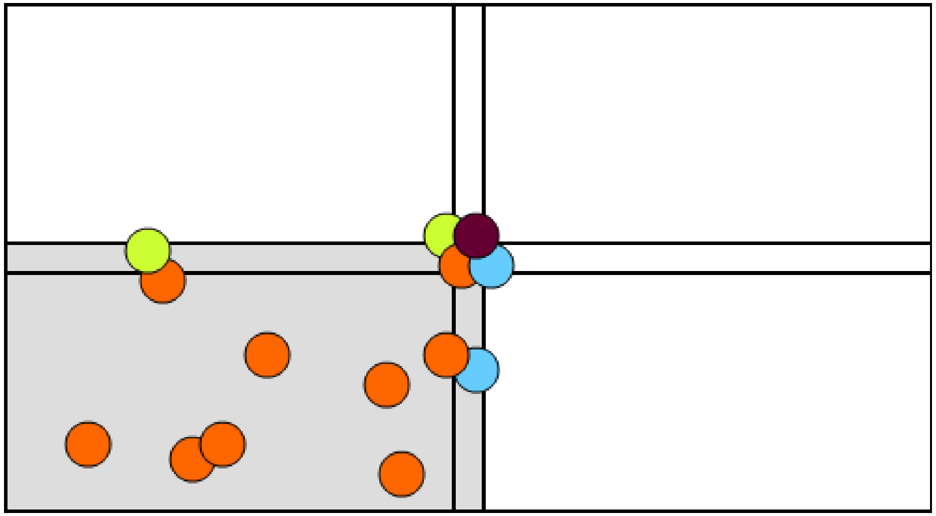


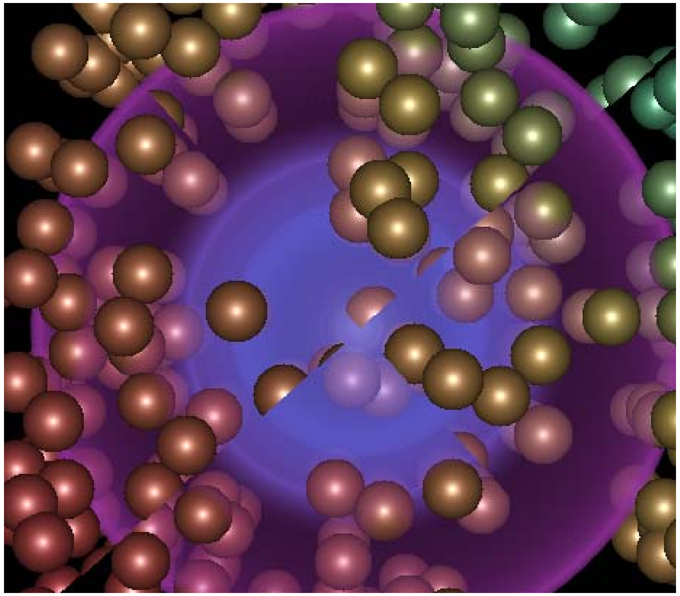
Parallel Rendering

- Particle Rendering
  - Software point sprite
    - Pre-calculated normal
    - Depth
    - Image space

Parallel Rendering

- Integrate Volume and Particle Rendering
  - Boundary issue





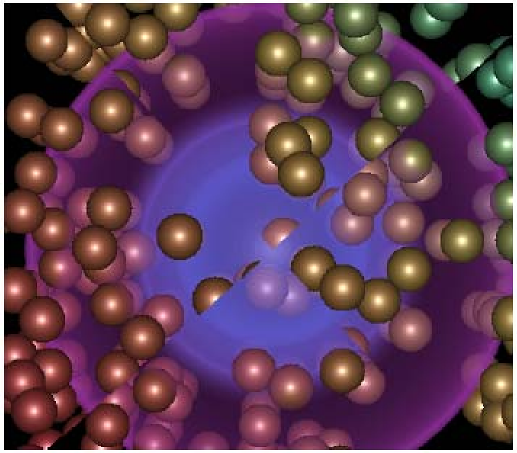
Parallel Rendering

- Integrate Volume and Particle Rendering

Algorithm Integrating volume rendering with particle rendering

---

1: render particles within its own data region;
2: exchange particles falling into the boundary with neighboring data regions;
3: render particles falling into the boundary from its own region and neighboring data regions;
4: read out the RGBA and the depth channels of particle rendering;
5: perform volume raycasting with depth lookup of the particle image for correct blending.



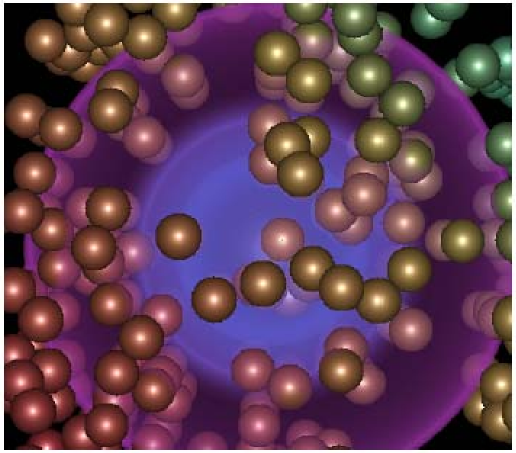


Image Compositing

- Direct Send
  - $N \cdot (N-1)$ messages exchanged among N PEs
  - Any number of processors

- Binary Swap
  - $N \cdot \log N$ messages exchanged among N PEs
  - Power-of-two processors

- 2-3 Swap
  - $O(N \cdot \log N)$ messages exchanged among N PEs
  - Any number of processors

Image Compositing

- 2-3 Swap
  - Multistage process
  - Partition processors into groups
  - 2-3 compositing tree
  - Scale well to thousands of processors

Integrating Visualization with Simulation

- Simulation Side

```
void s3drender_init_(
        int *myid, int *gcomm,
        double *species,
        char *speciesNames,
        double *loc,
        double *x, double *y, double *z,
        int *nx, int *ny, int *nz,
        int *npx, int *npy, int *npz,
        int neighbors[6])
```

MPI Communicator

pointer to local scalar variable

pointer to local particle data

size and coordinates of
global domain
and local partition

neighbor processors

Integrating Visualization with Simulation

- Visualization Side
  - Perform volume and particle rendering
  - Calculate and gather depth value
  - Visibility sorting
  - Build compositing tree
  - Image composting

Performance

- Test Environment
  - Cray XT5 at (NCCS), total 224,256 compute cores. Each node contains two hex-core AMD Opteron processors, 16GB memory, and a SeaStar 2+ router.

Performance

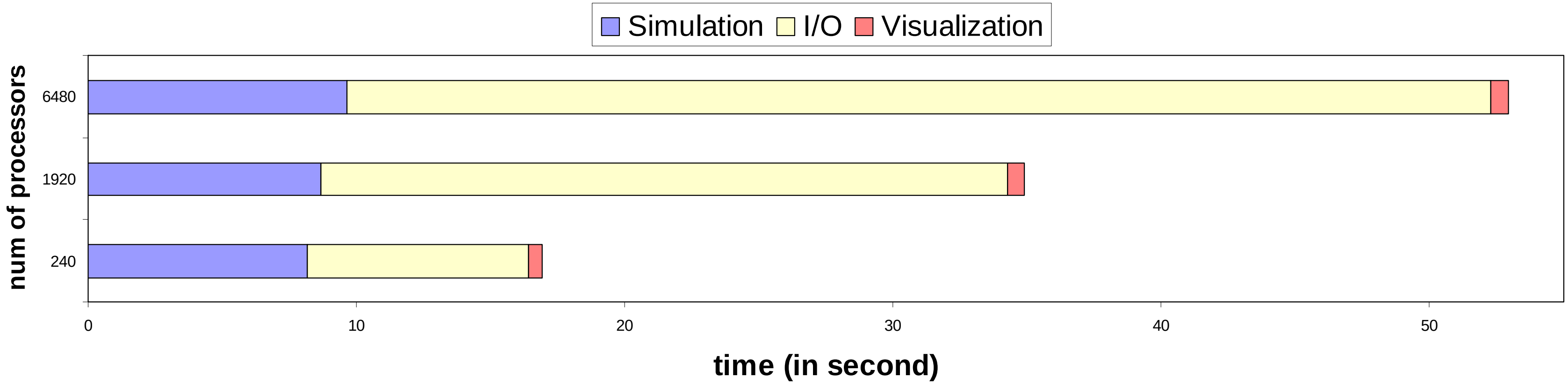
- Experiment
  - Simulation

# procs	240	1920	6480
volume size	$405 \times 320 \times 80$	$810 \times 640 \times 160$	$1215 \times 960 \times 240$
# variables	27	27	27
data size	$2.1GB$	$16.7GB$	$56.3GB$
# particles	$0.8M$	$5.2M$	$17.4M$
# variables	118	118	118
data size	$0.3GB$	$2.5GB$	$8.3GB$

  - Visualization
    - Image Resolution: 512^2, 1024^2 and 2048^2
    - Image Type: float, unsigned short and unsigned byte

Performance

Timing breakdown of simulation, I/O, and visualization for one time step



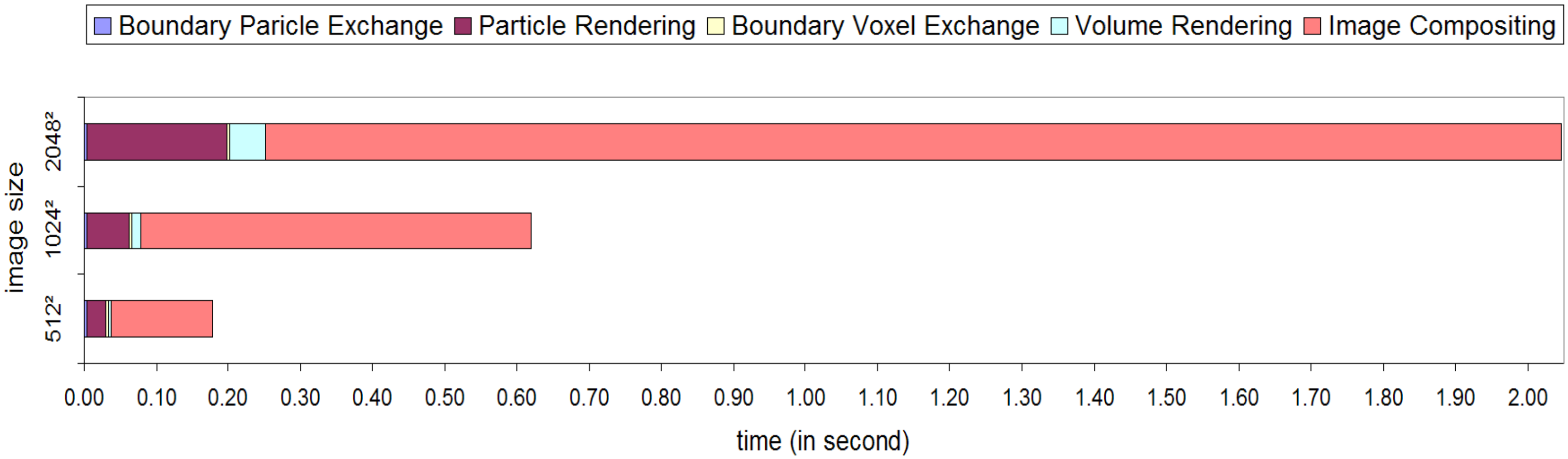
6480 processors, 1024^2 image resolution, and float image type:

Visualization time : ~ 6% of simulation time

I/O time : ~ 400% of simulation time

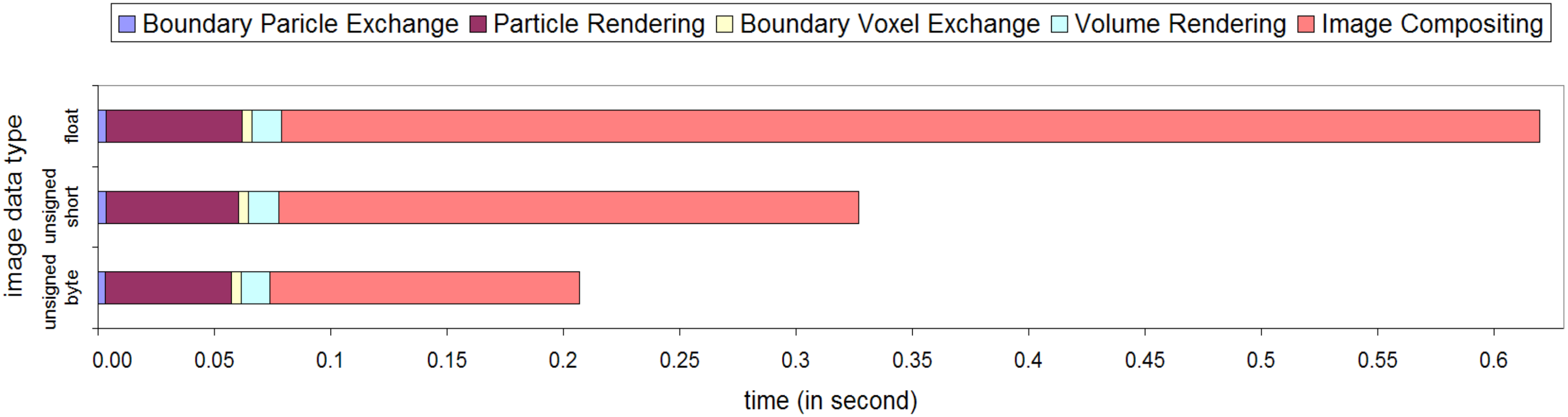
Performance

Timing breakdown of visualization for one time step with 1920 processors and float image type



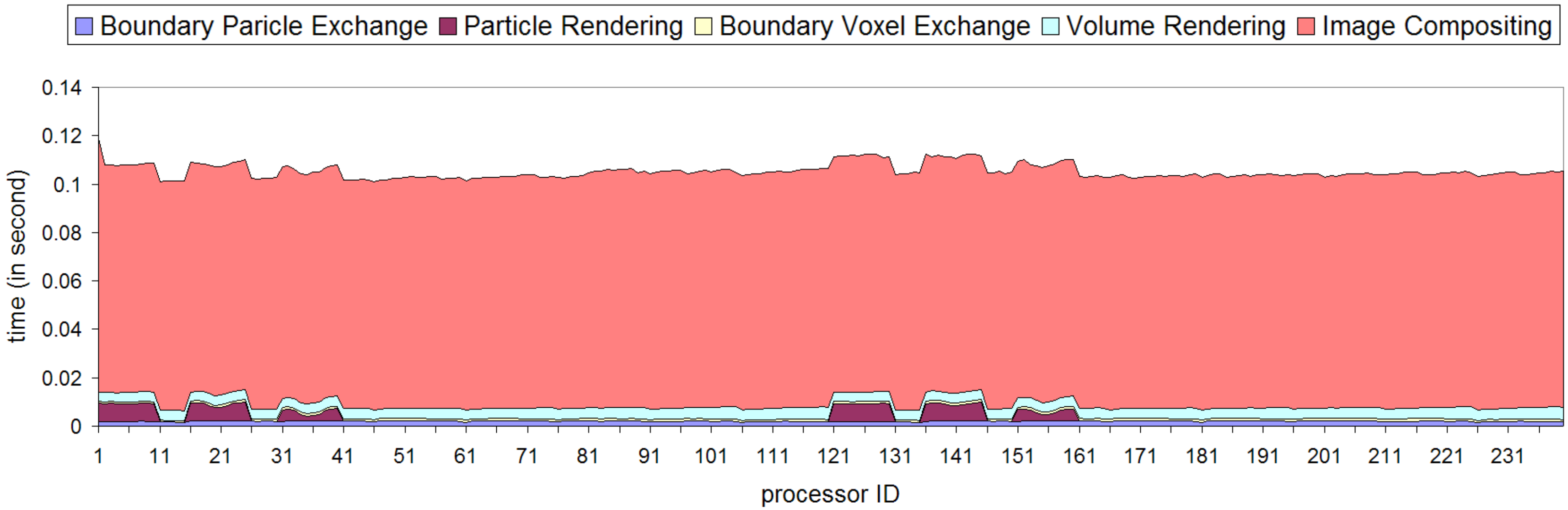
Performance

Timing breakdown of visualization for one time step with 1920 processors and 1024^2 image resolution



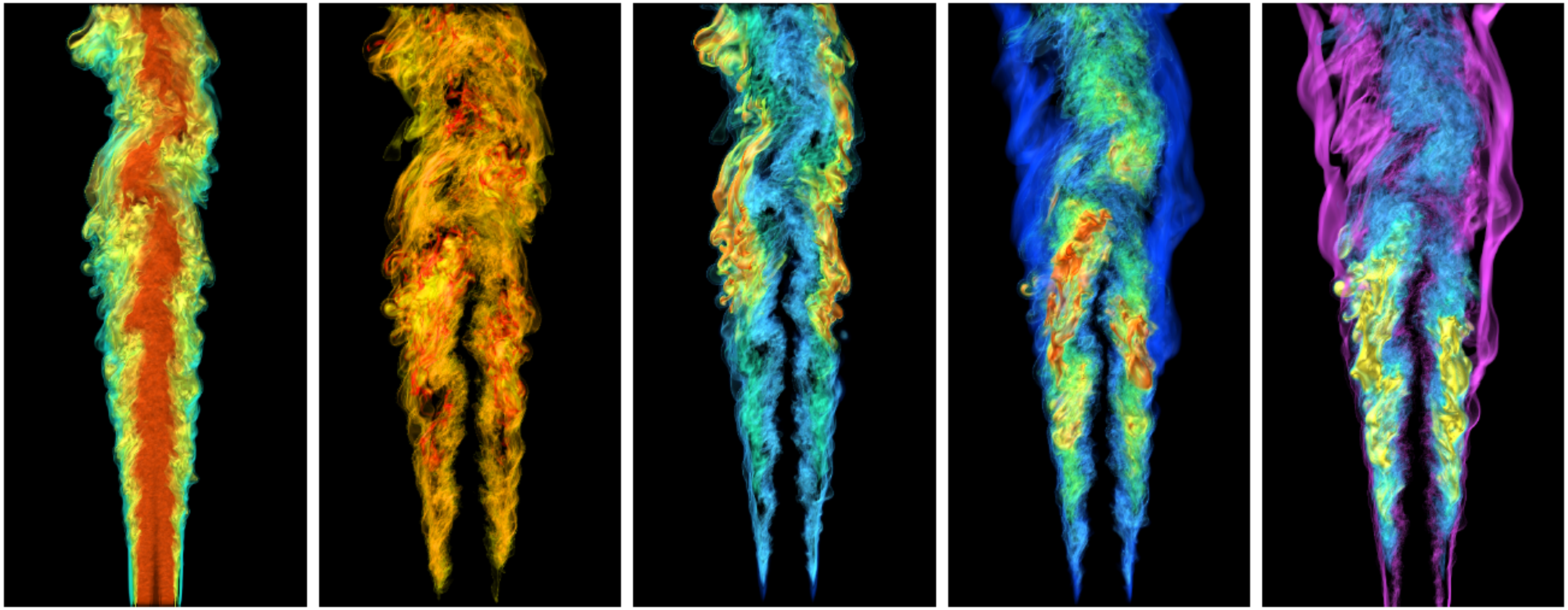
Performance

Timing breakdown of visualization for each processor with 240 processors and 512^2 image resolution



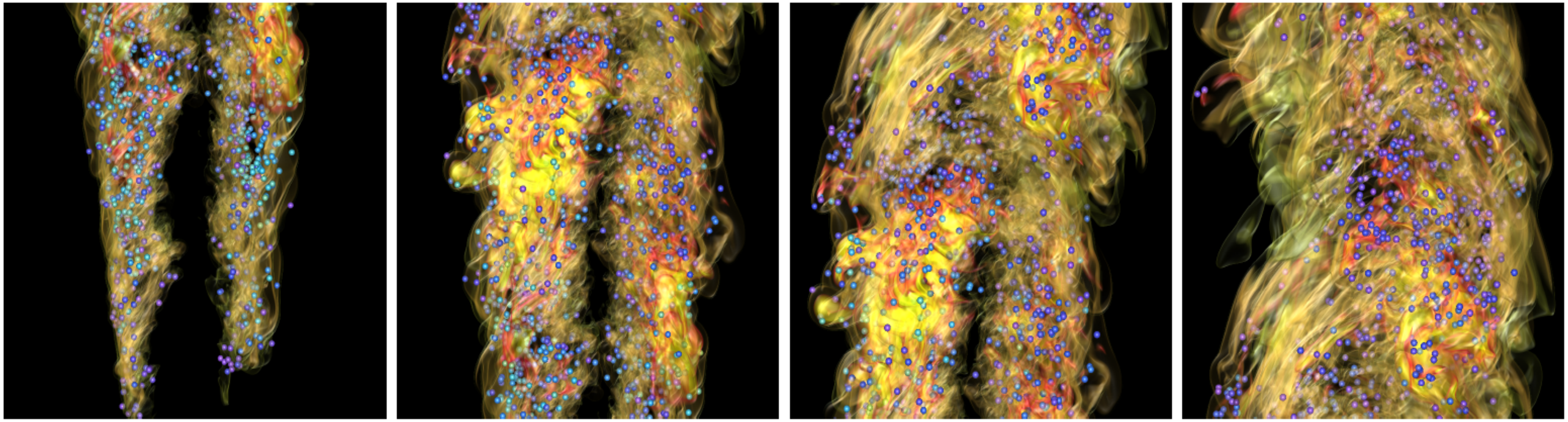
Results

- Volume rendering results of five selected variables : C2H4, CH2O, CH3, H2O2, HO2



Results

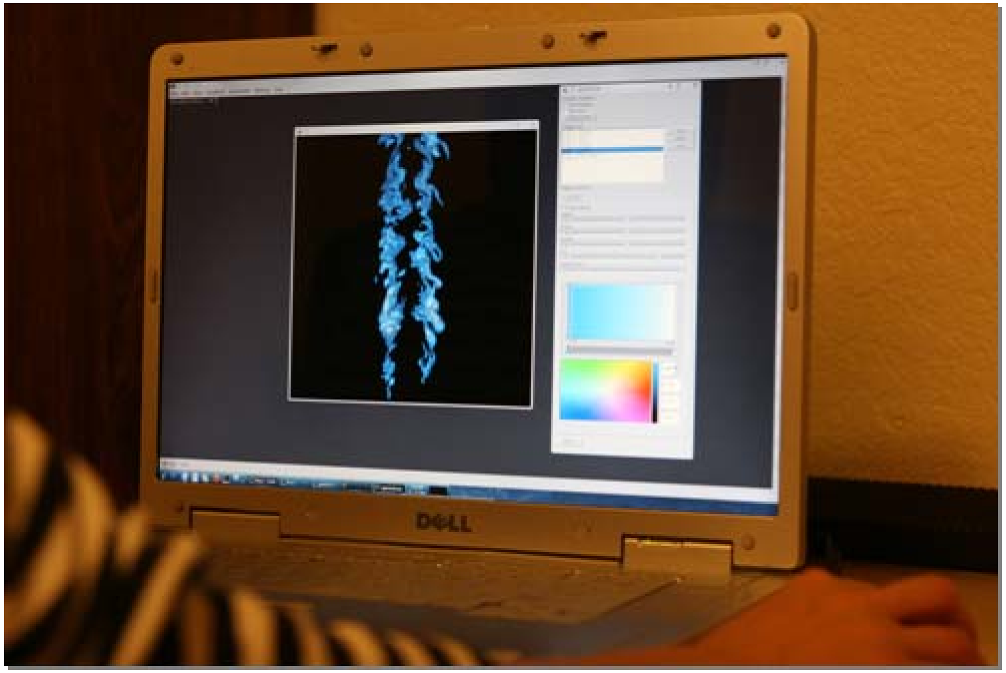
- Selected zoomed-in views of mix rendering of volume and particle data (volume variable CH2O and particle variable HO2)



Results

- Client program
  - Run on remote user's desktop/laptop and communicate with simulation over the network

- Demo
  - Screen capture from a laptop
  - Simulation runs on 2500 cores on XT5
  - Perform in-situ visualization every time step



[redacted]

Discussion

- Boundary Data
- Parallel Image Compositing
- Transfer Function and View Settings

Summary

- In-Situ Visualization
  - Use same computing platforms as simulations
  - Eliminate I/O and network bandwidth bound
  - Debug and monitor simulations
  - Study the full extent of the data

- Future Work
  - In-Situ Processing
    - Feature extraction
    - Data reduction

    ……

Acknowledgement

- US Department of Energy, Office of Advanced Scientific Computing Research and by the DOE Basic Energy Sciences Division of Chemical Sciences, Geosciences and Biosciences.
- DOE through the SciDAC program with Agreement No. DE-FC02-06ER25777, DOE-FC02-01ER41202,and DOE-FG02-05ER54817.
- Sandia National Laboratories is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under contract DE-AC04-94-AL85000.
- Supercomputing time provided by the National Center for Computational Sciences at Oak Ridge National Laboratory.

Thank You