LARGE SCALE VISUALIZATION ON GPU ACCELERATED SUPERCOMPUTERS

Peter Messmer, 11/16/2015

NVIDIA

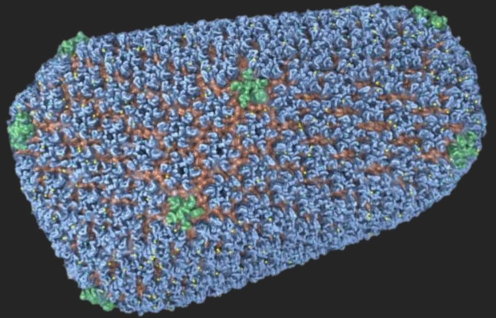
VISUALIZATION-ENABLED SUPERCOMPUTERS

CSCS Piz Daint

Galaxy formation

http://blogs.nvidia.com/blog/2014/11/19/gpu-in-situ-milky-way/

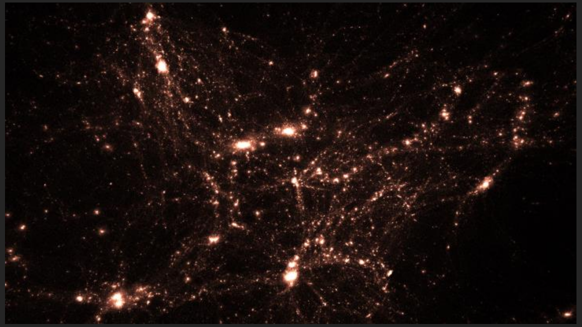
NCSA Blue Waters



Molecular dynamics

http://devblogs.nvidia.com/parallelforall/hpc-visualization-nvidia-tesla-gpus/

ORNL Titan



Cosmology

http://www.sdav-scidac.org/29-highlights/visualization/66-accelerated-cosmology-data-anal.html

SUPPORTING MULTIPLE VISUALIZATION WORKFLOWS

LEGACY WORKFLOW

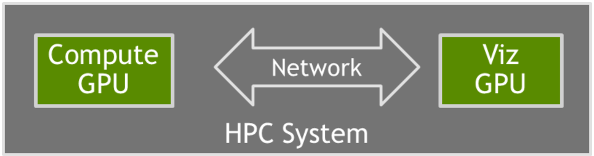
Separate compute & vis system

Communication via file system

PARTITIONED SYSTEM

Different nodes for different roles

Communication via high-speed network



CO-PROCESSING

Compute and visualization on same GPU

Communication via host-device transfers or memcpy

EGL CONTEXT MANAGEMENT

Leaving it to the driver

Top systems support OpenGL under X

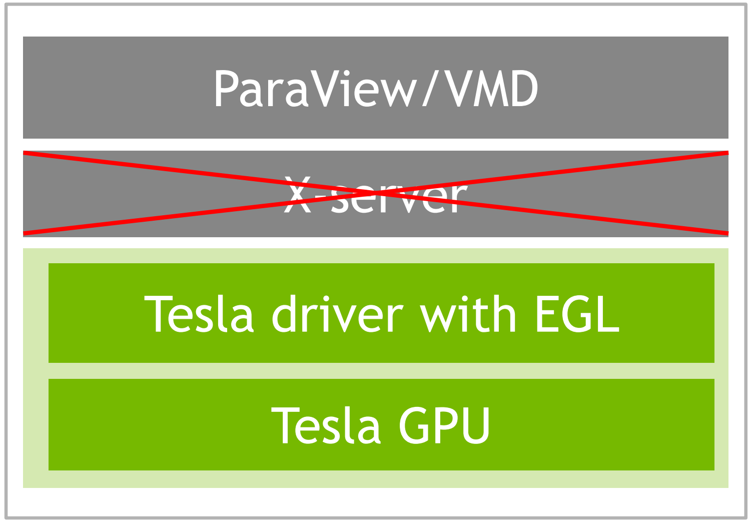
EGL: Driver based context management

Support for full OpenGL*, not only GL ES

Available in e.g. VTK

New opportunities for CUDA/OpenGL** interop

*Full OpenGL in r355.11; **CUDA interop in r358.7



EFFICIENT RENDERING AT SCALE

Modern networks remove compositing bottleneck

Sort last compositing perceived bottleneck

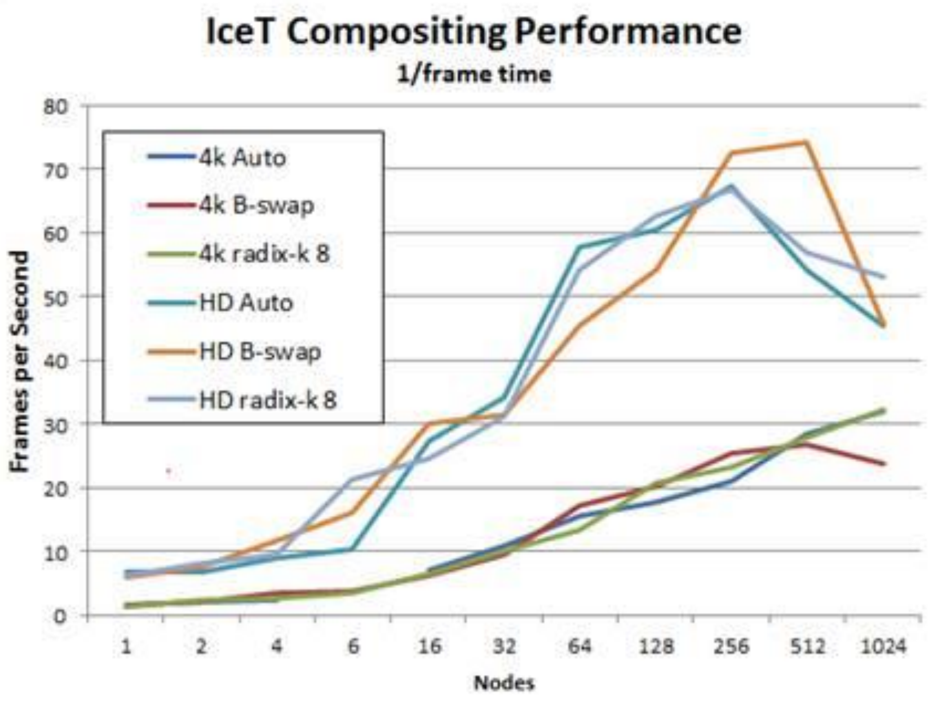
Today: fast networks, pipelining and novel algorithms

> 30 fps on 4k frames on 1024 nodes possible

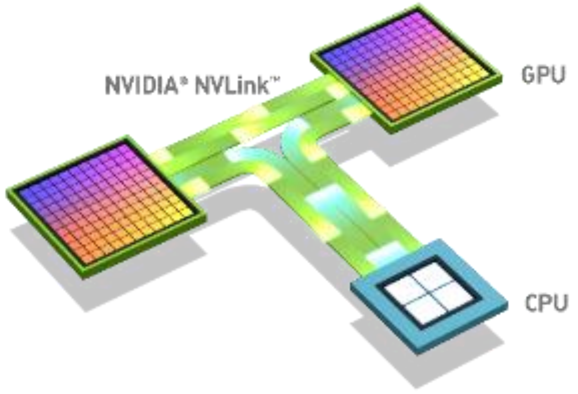
Enables real-time viz at large concurrency

Enables very large geometries

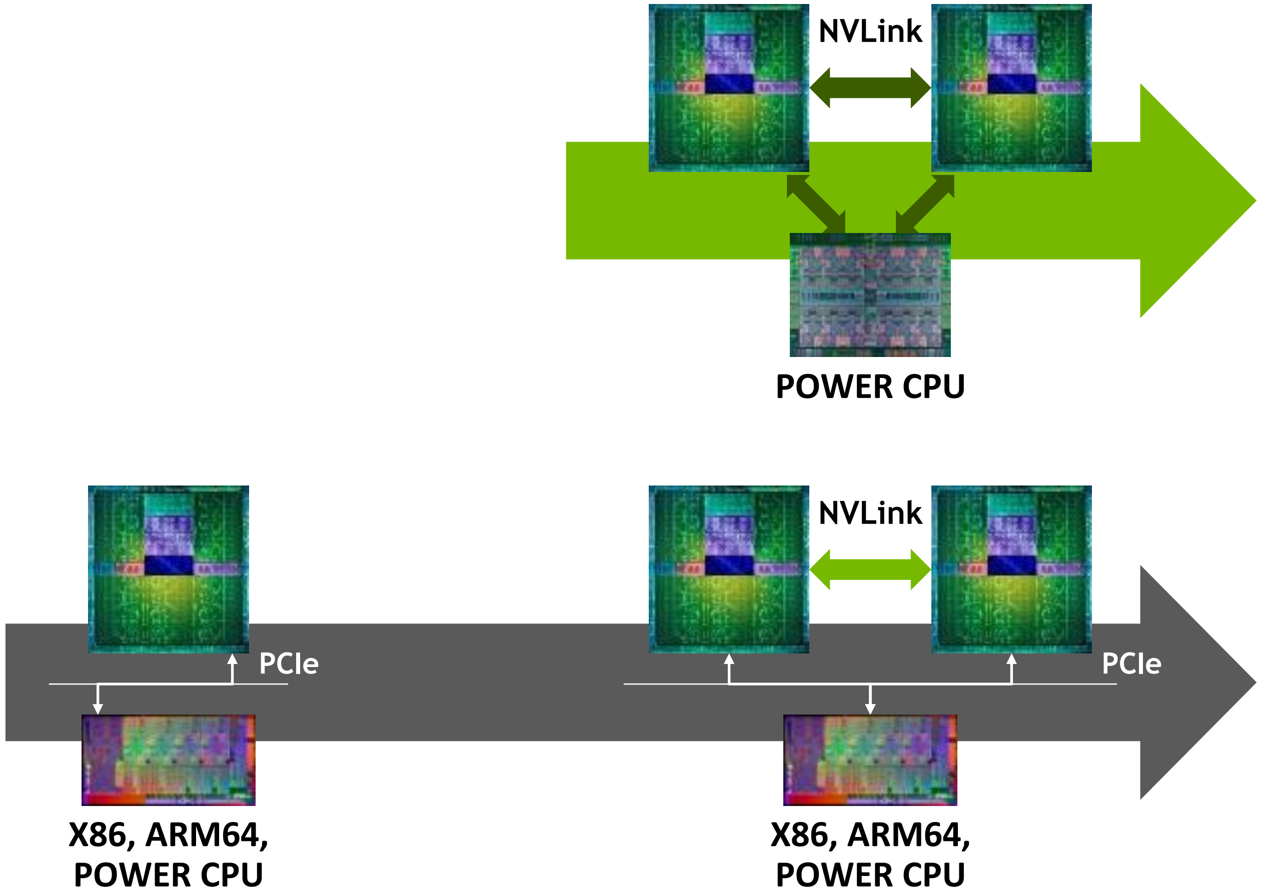
(e.g. Piz Daint: 30 TB of GPU memory)



NVLINK
HIGH-SPEED GPU INTERCONNECT



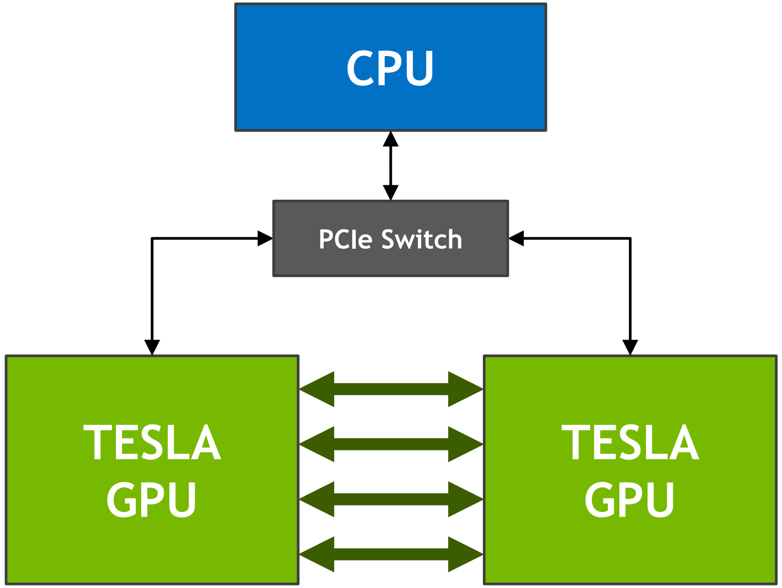
KEPLER GPU	PASCAL GPU



2014	2016

NVLINK UNLEASHES MULTI-GPU PERFORMANCE

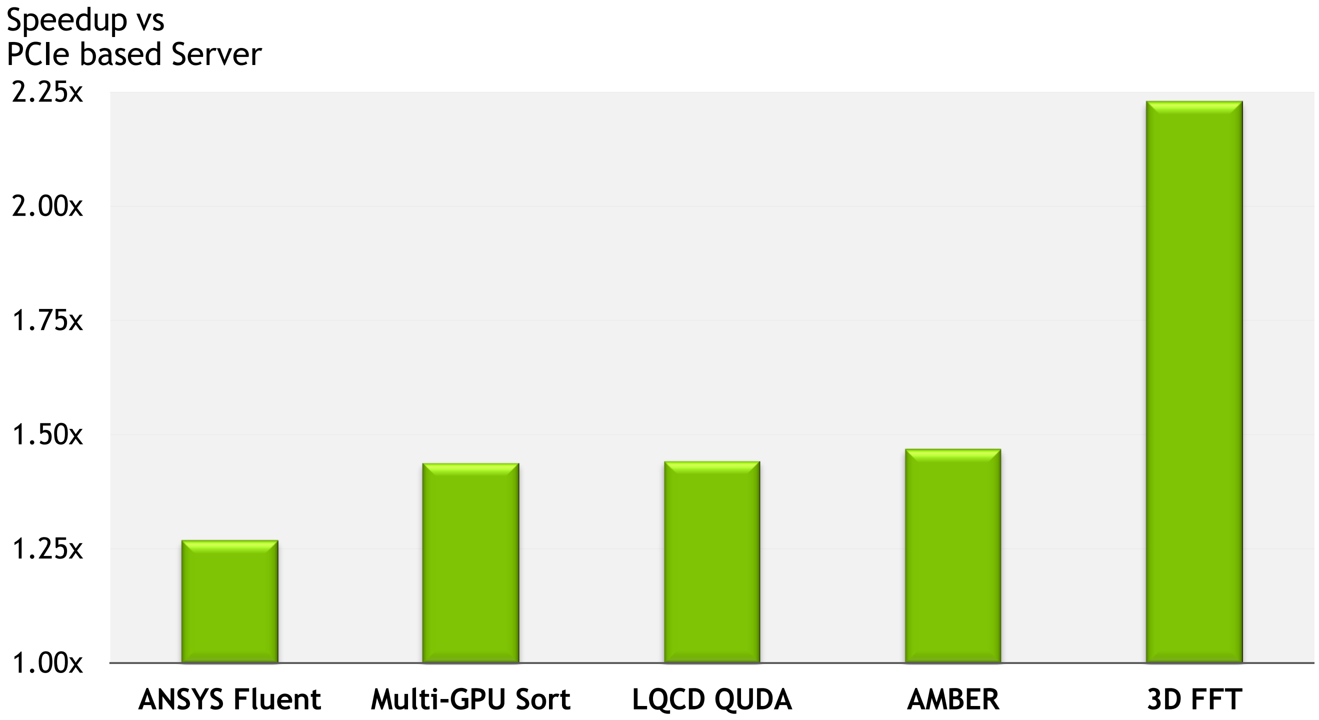
GPUs Interconnected with NVLink



5x Faster than
PCIe Gen3 x16

Over 2x Application Performance Speedup

When Next-Gen GPUs Connect via NVLink Versus PCIe



3D FFT, ANSYS: 2 GPU configuration, All other apps comparing 4 GPU configuration
AMBER Cellulose (256x128x128), FFT problem size (256^3)

CUDA

Super Simplified Memory Management Code

CPU Code

```
void sortfile(FILE *fp, int N) {
  char *data;
  data = (char *)malloc(N);

  fread(data, 1, N, fp);

  qsort(data, N, 1, compare);


  use_data(data);

  free(data);
}
```

CUDA 6 Code with Unified Memory

```
void sortfile(FILE *fp, int N) {
  char *data;
  cudaMallocManaged(&data, N);

  fread(data, 1, N, fp);

  qsort<<<...>>>(data,N,1,compare);
  cudaDeviceSynchronize();

  use_data(data);

  cudaFree(data);
}
```

OpenACC

Simple | Powerful | Portable

Fueling the Next Wave of

Scientific Discoveries in HPC

```
main()
{
  <serial code>
  #pragma acc kernels
  //automatically runs on GPU
  {
    <parallel code>
  }
}
```

University of Illinois
PowerGrid- MRI Reconstruction

70x Speed-Up
2 Days of Effort

RIKEN Japan
NICAM- Climate Modeling

7-8x Speed-Up
5% of Code Modified

8000+

Developers

using OpenACC

http://www.cray.com/sites/default/files/resources/OpenACC_213462.12_OpenACC_Cosmo_CS_FNL.pdf
http://www.hpcwire.com/off-the-wire/first-round-of-2015-hackathons-gets-underway
http://on-demand.gputechconf.com/gtc/2015/presentation/S5297-Hisashi-Yashiro.pdf
http://www.openacc.org/content/experiences-porting-molecular-dynamics-code-gpus-cray-xk7

MODERN OPENGL FOR HPC VIZ

Mandatory to access advanced rendering features

VTK supports now OpenGL 3.2

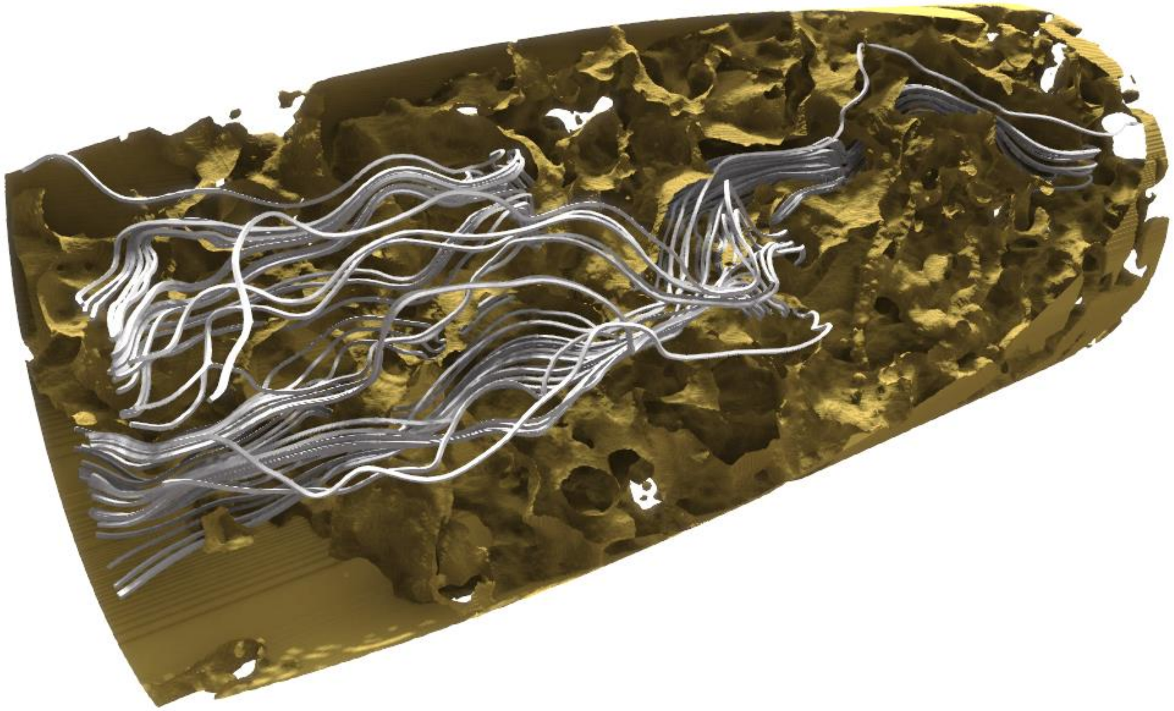
Access to new shaders (AO, VXGI, ..)

Some algorithms well suited for

distributed memory rendering

GPU hardware support

Multi-casting for VXGI



HIGH FRAMERATE = MINIMAL IMPACT ON SIMULATION

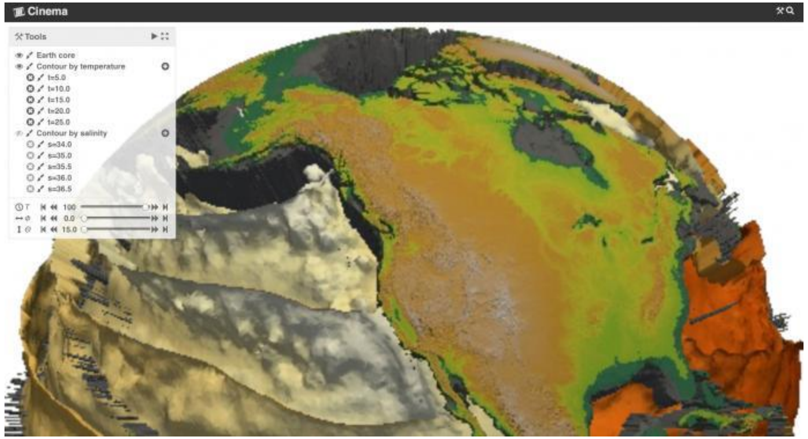
FPS matter, even in HPC

Real-time visualization only one use case

Batch processing will not immediately disappear

Acceptable time budget for visualization/analysis

More diagnostics in the same time



ParaView Cinema

ACCELERATED REMOTE RENDERING WITH VIDEO ENCODING

Interactivity over large distances

Lossy and loss-less (Maxwell +) H264 encoder

Separate unit, does not consume “GPU resources”

Leveraged by commercial, free tools

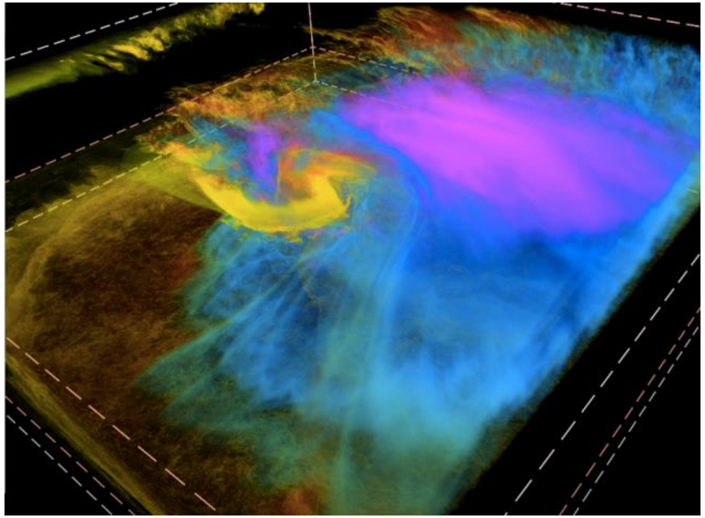
Available on e.g. Titan

Possible use for non-video data

https://developer.nvidia.com/nvidia-video-codec-sdk

SCALABLE RENDERING AND COMPOSITING

NVIDIA INDEX



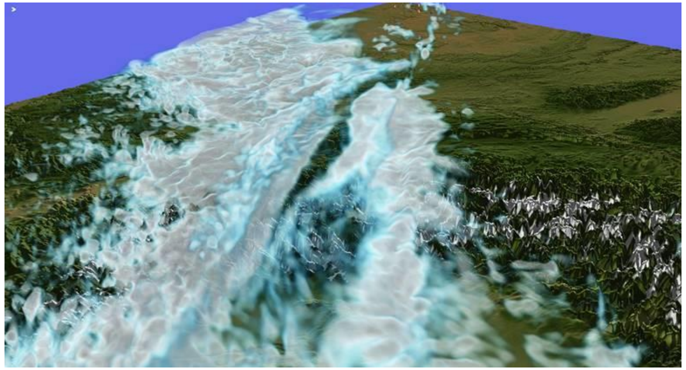
Large-scale (volume) data visualization

Interactive visualization of TB of data

Stand-alone or coupling into simulation

HW Accelerated remote rendering

Plugin for ParaView



http://www.nvidia-arc.com/products/nvidia-index.html

NVIDIA INDEX FOR PARAVIEW

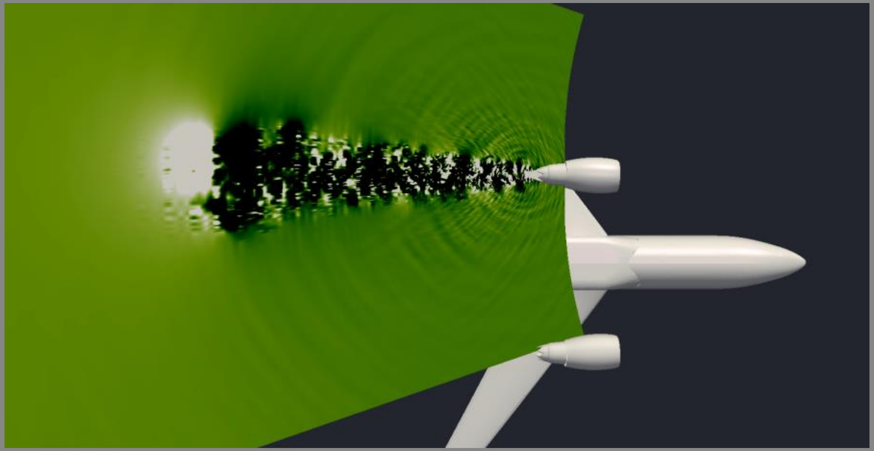
Scalable volume rendering solution in ParaView for large data (Evaluation version available in Q1 2016)

Uses GPU clusters to deliver interactivity performance needed by scientists

"I was very impressed with the responsive performance and high quality volume rendering of NVIDIA IndeX for ParaView on terabytes of data from my large thunderstorm simulation. Being able to interact with the full dataset in real-time is tremendously useful to me in uncovering science that is not currently possible with other solutions."

- Dr. Leigh Orf
U. of Wisconsin-Madison

IN-SITU VISUALIZATION ON TITAN



First prototype of ParaView in-situ visualization capabilities in pyFR (CFD) simulations, predicting jet engine acoustics

Both compute and visualization running on Titan GPUs and streaming to a remote location

"When running PyFR at scale, it generates very large data sets that need analyzing for acoustics. The traditional post hoc method is simply not fit for purpose – in situ visualization and processing are critical. We see a potential for 50x speed ups with in situ, which significantly accelerates our scientific discovery"

- Dr. Peter Vincent
Imperial College

VISUALIZATION ON TESLA

Efficiency

- HW accelerated rendering
- Remoting support
- Simulation interop
- Maximized data locality

Fidelity

- Advanced rendering algorithms
- Improved perception
- Faster feedback

Flexibility

- Scalable visualization
- Multiple configurations for viz+sim

VISUALIZATION ON GPU ACCELERATED SUPERCOMPUTERS

GPU accelerated supercomputers support different visualization workflows

Filter and render on GPU

Use of hardware accelerated OpenGL features simplified by EGL

Fast compositing enables efficient distributed memory rendering at high frame rate or minimal overhead

Compression hardware enables image delivery at high frame rates

Use of advanced OpenGL in tools enable novel capabilities (often with GPU support)

NVLink simplifies locality management

Join the discussion @ SC15:
BOF on Interactivity in HPC
Tuesday, 11/17, 5:30pm

NVIDIA