Responsive Large Data Analysis and Visualization with the ParaView Ecosystem

Patrick O’Leary, Kitware Inc

Hybrid Computing

Attribute	Titan	Summit - 2018
Compute Nodes	18,688	~3,400
Processor	(1) 16-core AMD Opteron per node	(Multiple) IBM POWER 9s per node
Accelerator	(1) NVIDIA Kepler K20x per node	(Multiple) NVIDIA Volta GPUs per node
Memory per node	32GB (DDR3)	>512GB (HBM+DDR4)
CPU-GPU Interconnect	PCI Gen2	NVLINK (5-12x PCIe3)
System Interconnect	Gemini	Dual Rail EDR-IB (23 GB/s)
Peak Power Consumption	9 MW	10 MW

It’s Hard

Unfortunately, hybrid computing systems are not without drawbacks. Developing and deploying an application on a hybrid system is more challenging than traditional clusters due to the heterogeneous nature of the system.

- The model of the computer taken when developing applications is often resource agnostic.
- Many applications are developed with a mostly unrestricted view of memory, which creates problems porting code to components with a restricted working set.
- The interconnect between the traditional and the non-traditional component(s) maybe high latency or low bandwidth, which may cause bottlenecks not originally present.
- The number of computational units sharing an interconnect is greatly increased in some cases, which can also lead to link saturation.
- Finally, some computing components require more explicit description of the fine- and coarse-grain parallelism.

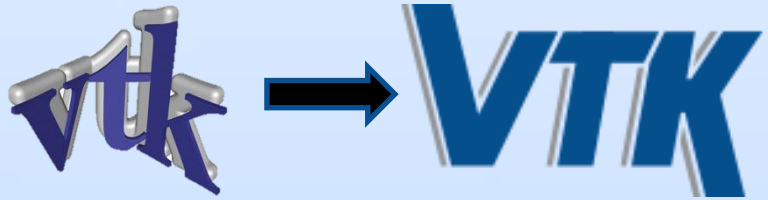
All of these issues transform a complex development task in to a formidable Herculean task.

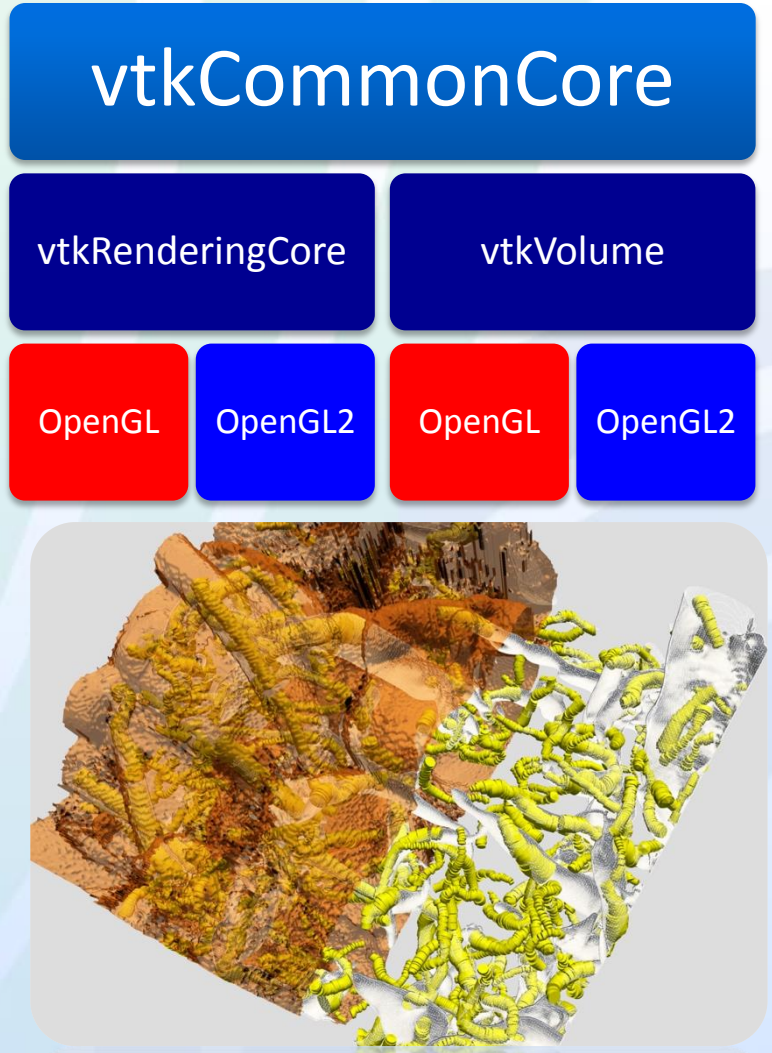
- A software company creating open-source collaboration platforms, which are used globally for
  - research
  - education
  - commercial applications
- This software is created by
  - internationally recognized experts
  - in extended communities
  - using a rigorous, quality-inducing software development process.

Note: *Inc. Magazine announced the 2012 Inc. 500|5000 list, an exclusive ranking of the nation's fastest-growing private companies. Kitware ranked #1245 overall and #100 among software companies, with three year revenue growth of 248%.*

VTK Update

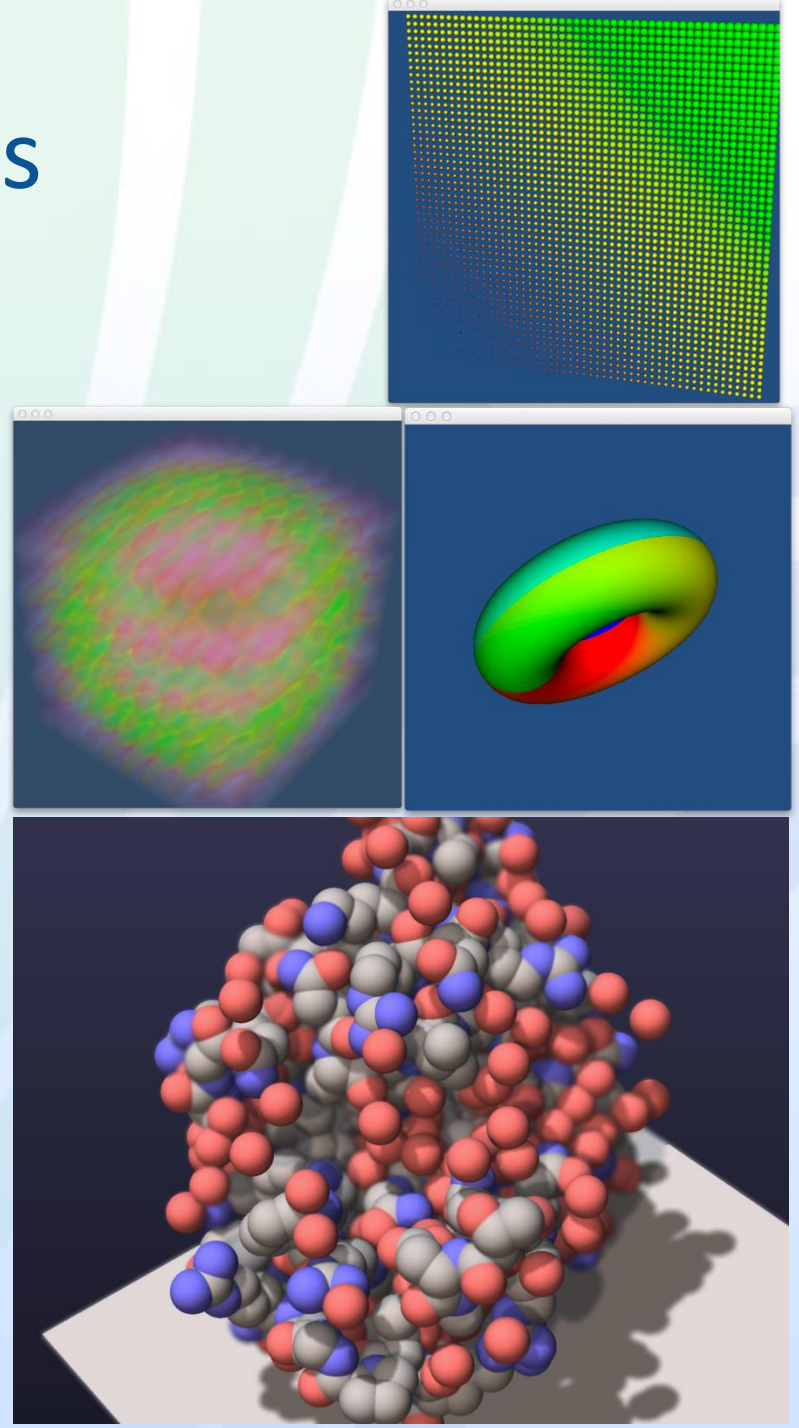
- NVIDIA and Kitware collaboration
- Major rewrite of VTK's rendering code
  - Mainly geometry and volume rendering
  - Rendering code for desktop and embedded systems
  - Take advantage of modern programmable GPU: migration from fixed pipeline to buffers and GLSL
  - Recast interaction with data pipelines to take full advantage of batching for improved transfer





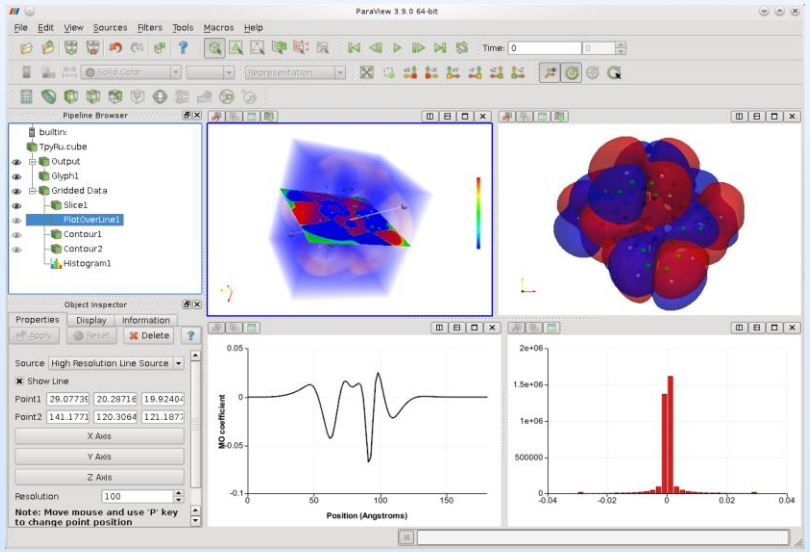
Performance, Scale and Shiny Pictures

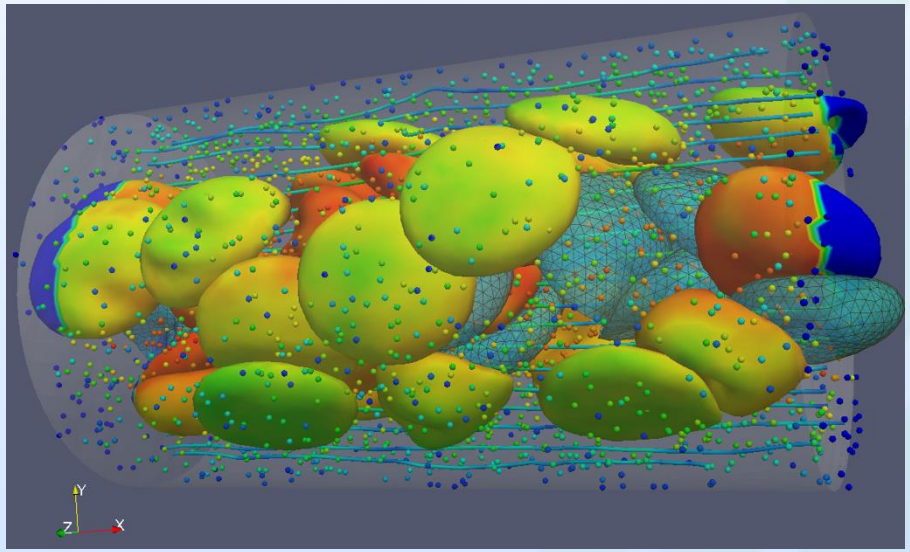
- Geometry rendering about 100 times faster
  - Initial frame/updated input ten times faster
  - Memory footprint about half for single large geometry
- Volume rendering twice as fast
  - Improved capability to run across systems
- Capable of rendering much larger systems
  - Single node/workstation rendering on 200M+ triangles
  - Client-server rendering sees similar per-node gains
- Other advanced rendering techniques
  - Glyphs, impostors, Gaussian splatting for data
- Main thread was CPU bound, now GPU bound

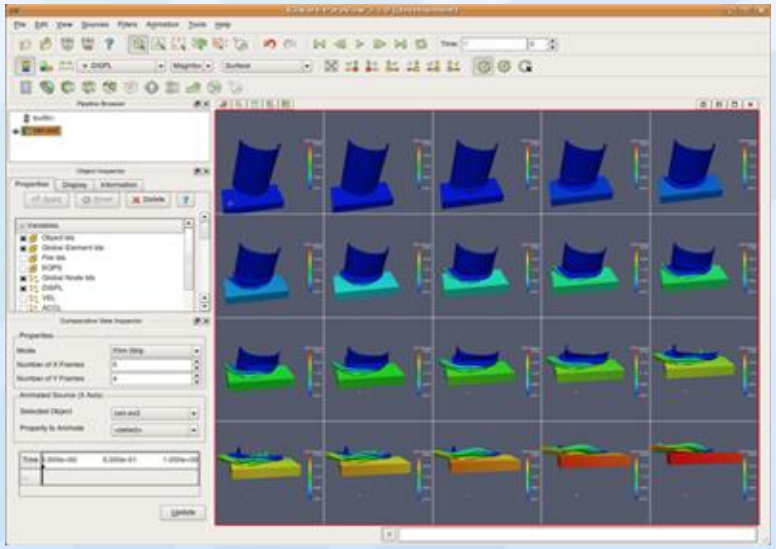


ParaView

ParaView is an **open-source**, **multi-platform**, **data analysis and visualization** application for analyzing **extremely large datasets** using distributed memory computing resources







History

Kitware

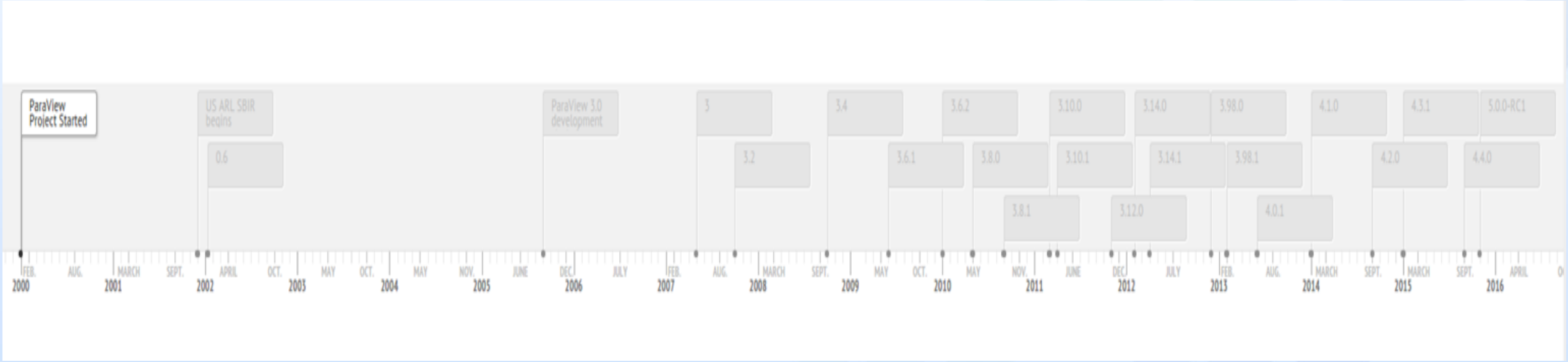
Sandia National Laboratories

Los Alamos NATIONAL LABORATORY EST. 1943

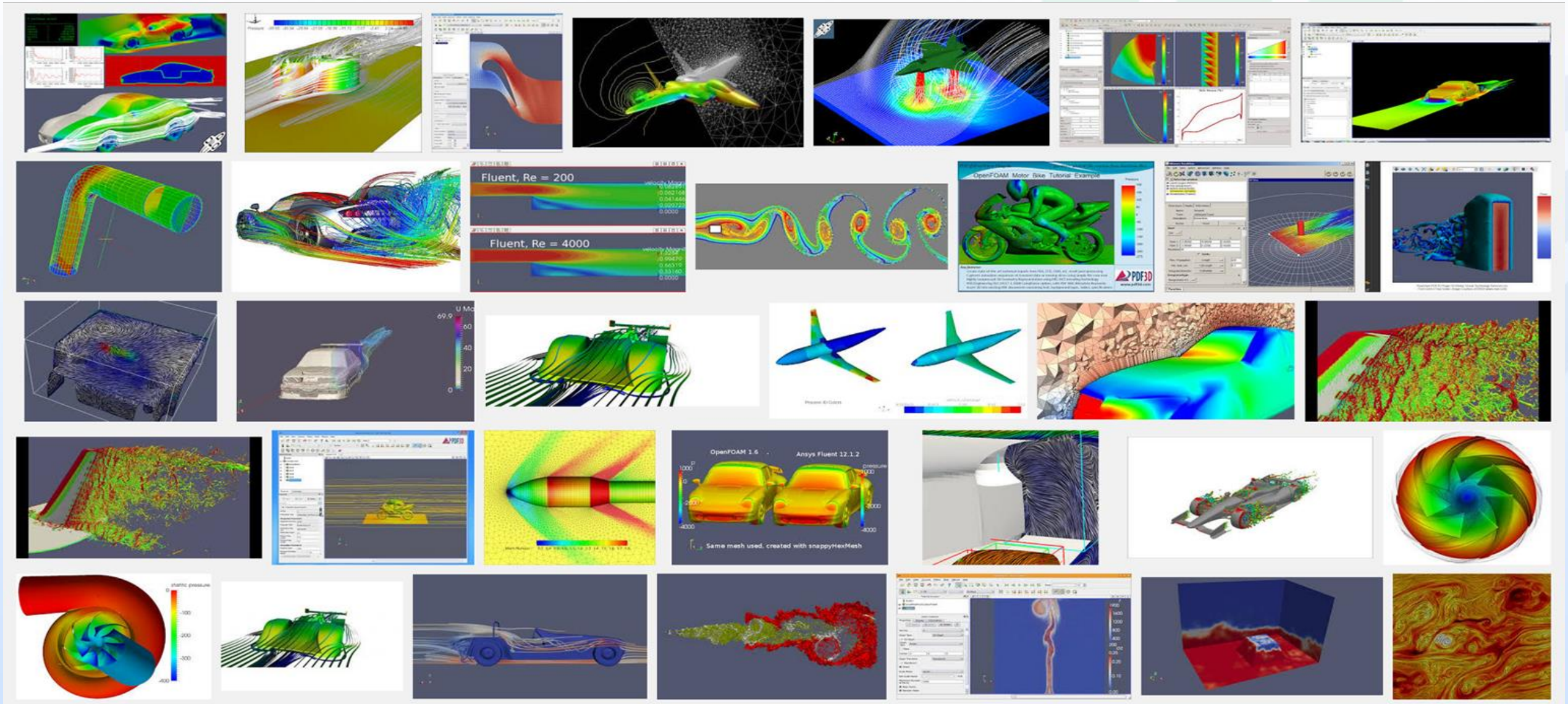
ASC™

U.S. Army Research Laboratory

and several more!



Data Analysis & Visualization



ParaView Catalyst

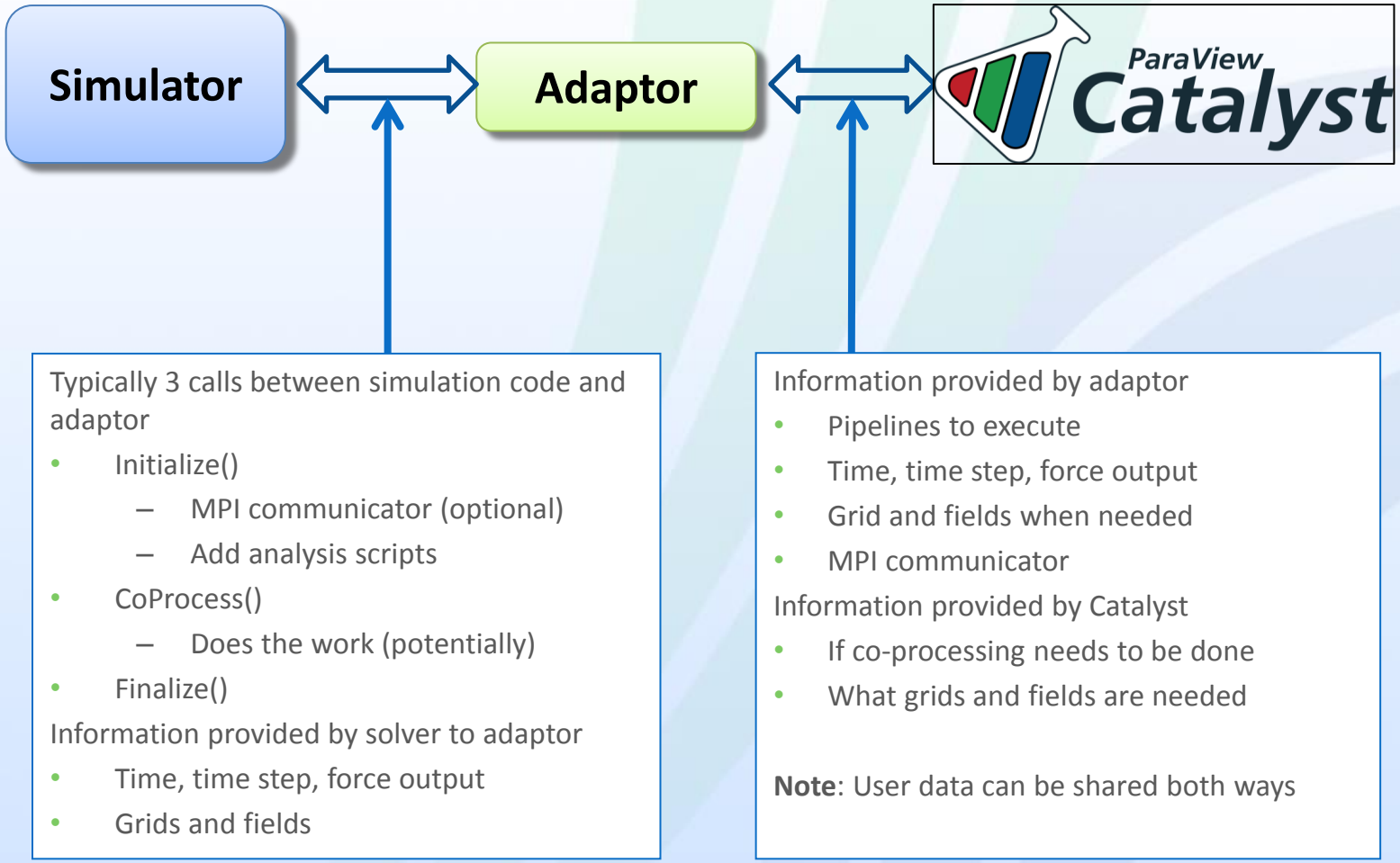
What is it?

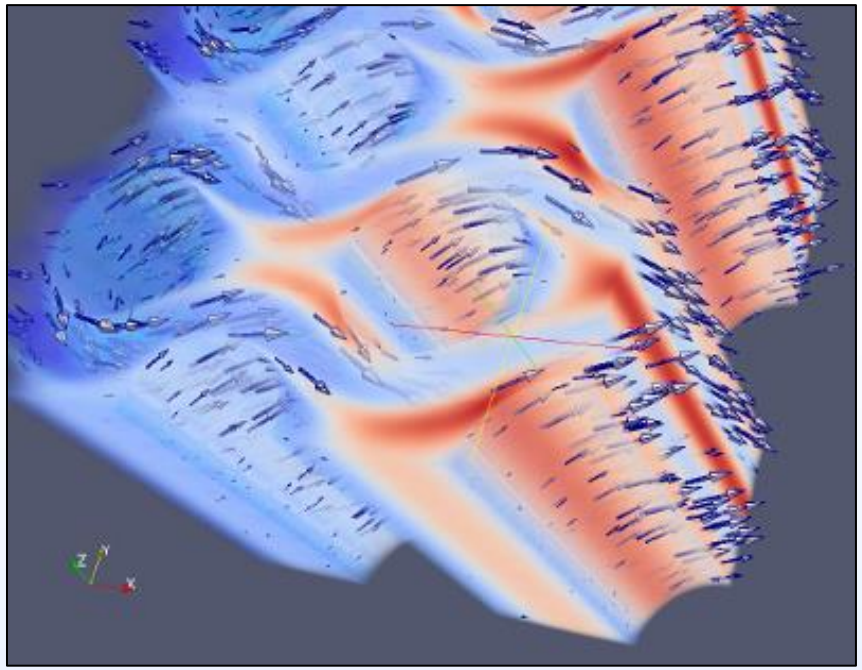
A set of tools for *in situ* analysis and visualization

Why use it?

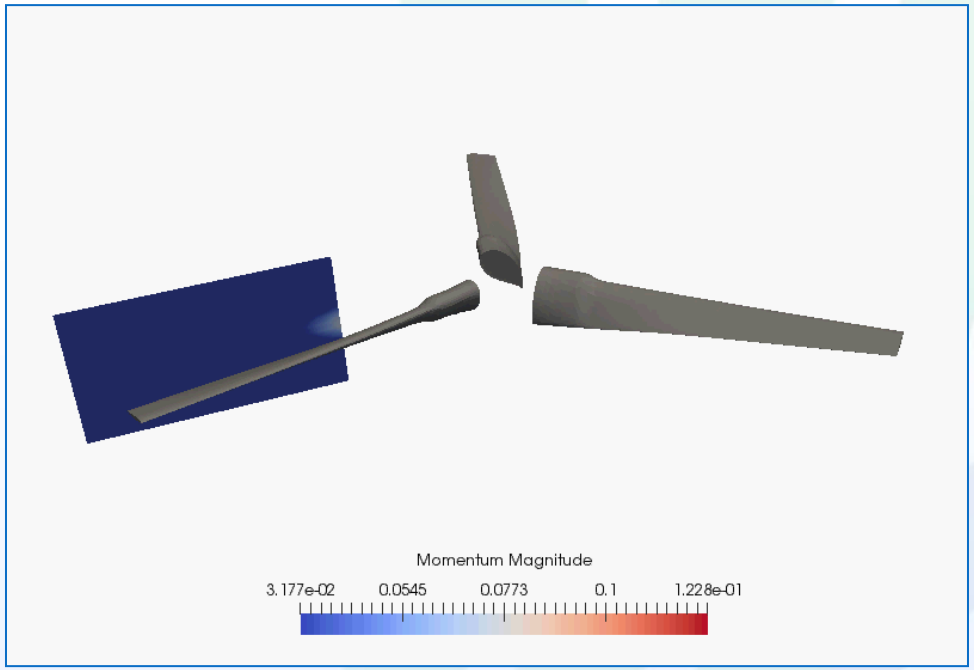
Brief answer – one solution to reduced relative IO performance on the road to exascale

More details – provide flexible analysis options and better usage of computational resources to reduce scientists' time to insight into given problems

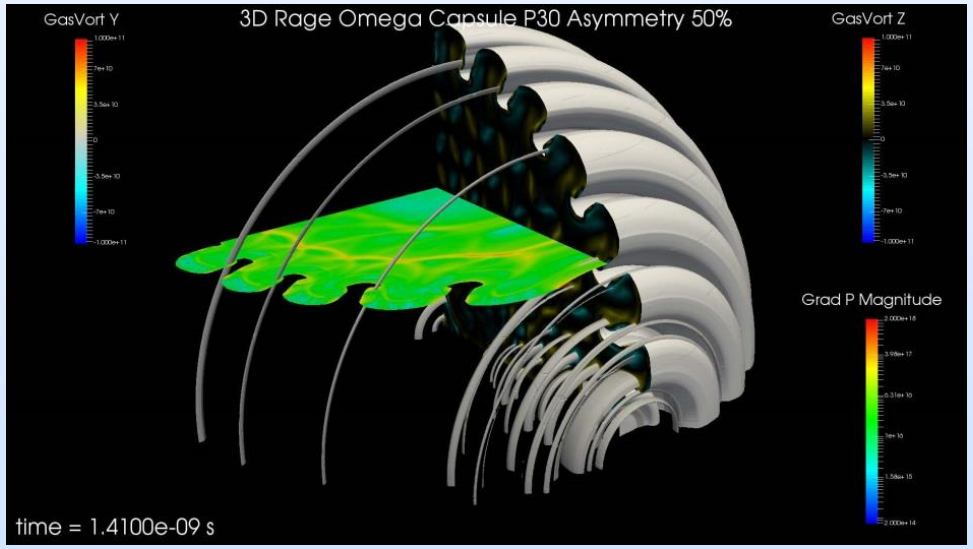




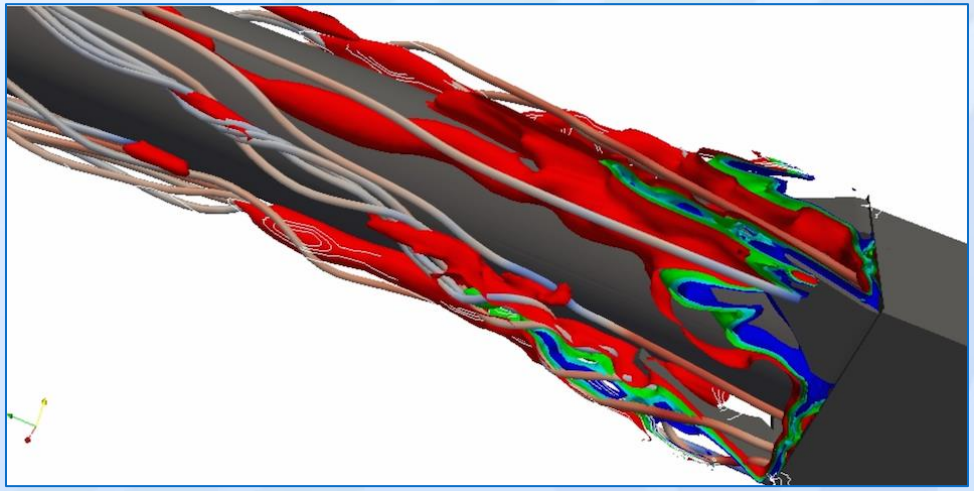
Code Saturne (EDF)



HPCMP CREATE-AV Helios (Army)



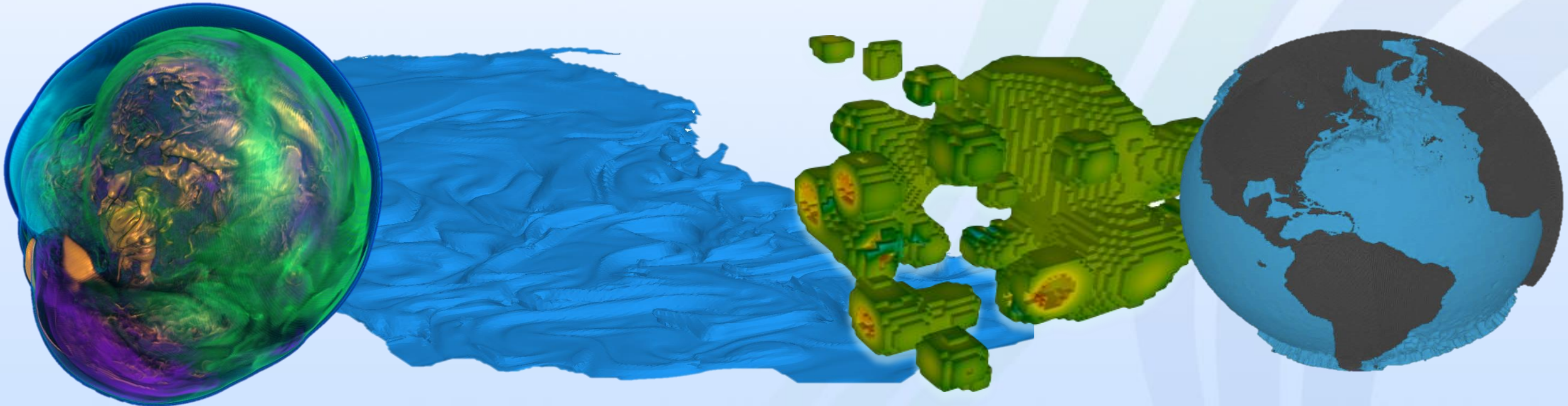
RAGE (LANL)



Hydra-TH (LANL)

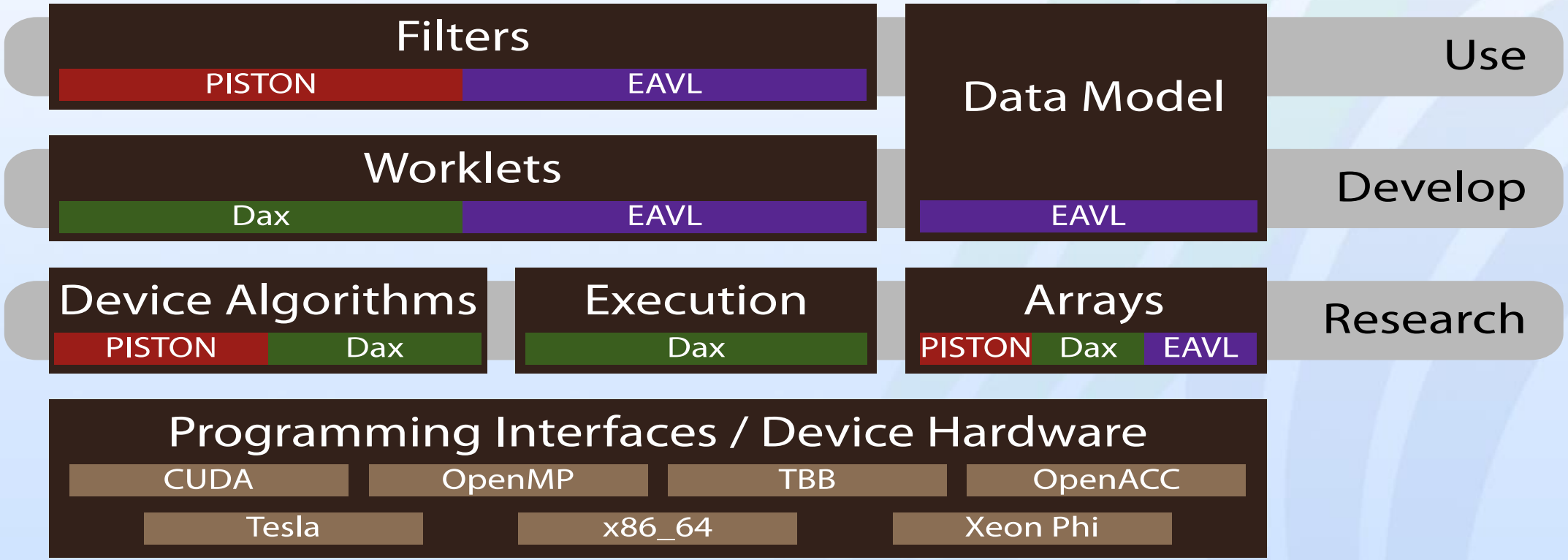
VTK-m

- A single place for the analysis and visualization community to collaborate, contribute, and leverage massively threaded algorithms
- Make it easier for simulation codes to take advantage of these parallel analysis and visualization and algorithms on all next-generation hardware
- Data parallel primitives provide an abstraction layer between the hardware's architecture and the high-level algorithm



VTK-m

- Brings together the development teams of multiple projects
  - EAVL, Oak Ridge National Laboratory
  - DAX, Sandia National Laboratory
  - PISTON, Los Alamos National Laboratory
- Working with hardware vendors to make VTK-m efficient



ParaView ‘Live’

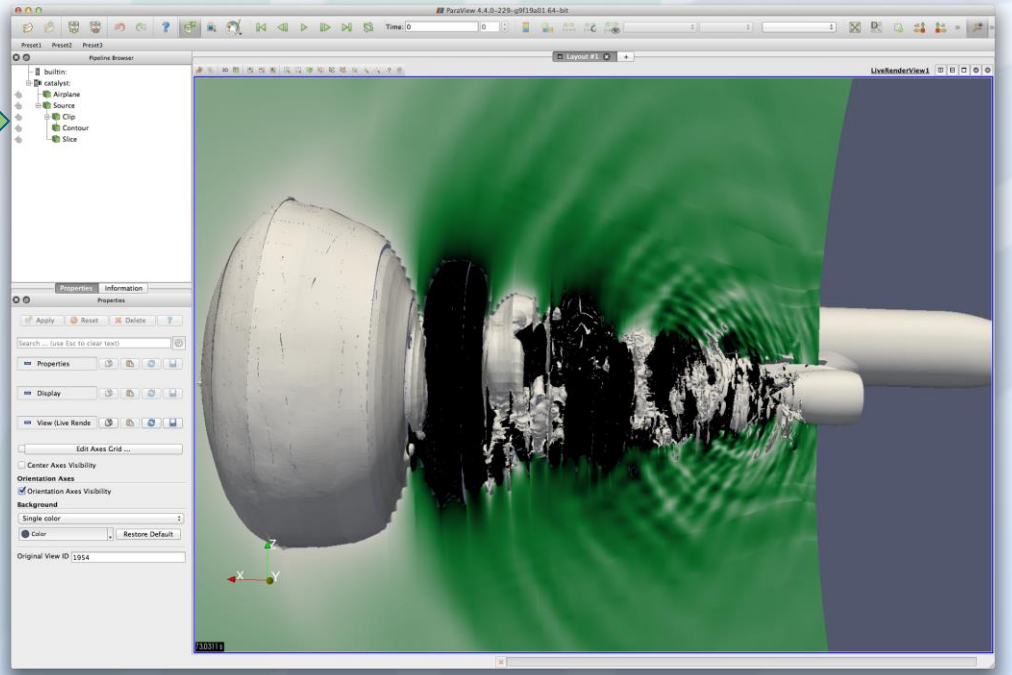
simulate

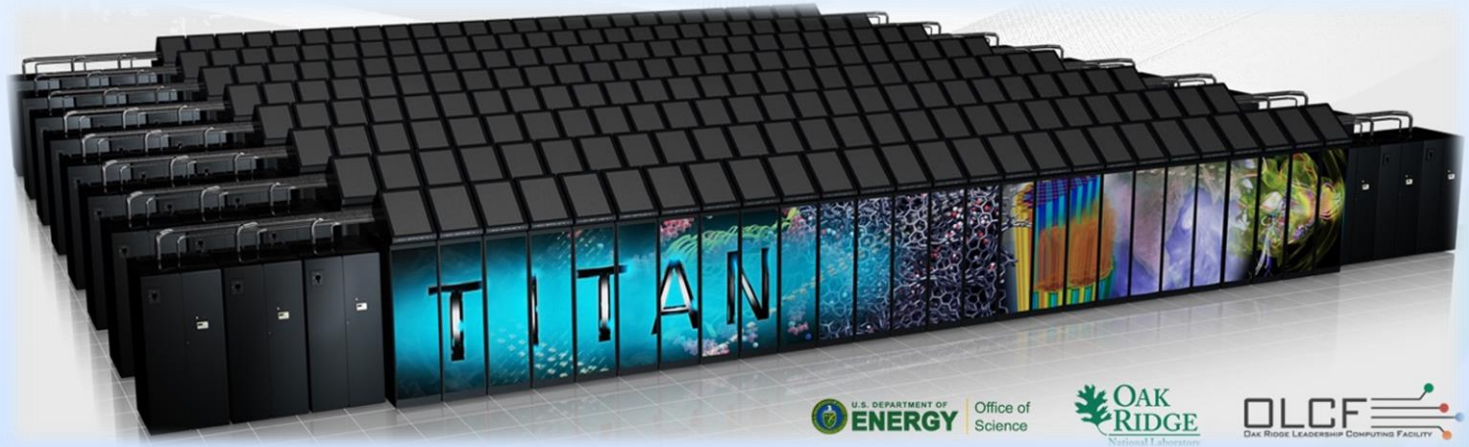
t = 0, 1, ….

ParaView Catalyst VTK-m

analyze + visualize

Extracts / Images





Acknowledgements

Imperial College @imperial.ac.uk

- Arvind Iyer (a.iyer)
- Jin Seok Park (jin-seok.park)
- Brian Vermeire (b.vermeire)
- Peter Vincent (p.vincent)
- Freddie Witherden (freddie.witherden08)

Kitware, Inc @kitware.com

- Utkarsh Ayachit (utkarsh.ayachit)
- T.J. Corona (tj.corona)
- David DeMarle (dave.demarle)
- Berk Geveci (berk.geveci)
- Robert Maynard (robert.maynard)

NVIDIA @nvidia.com

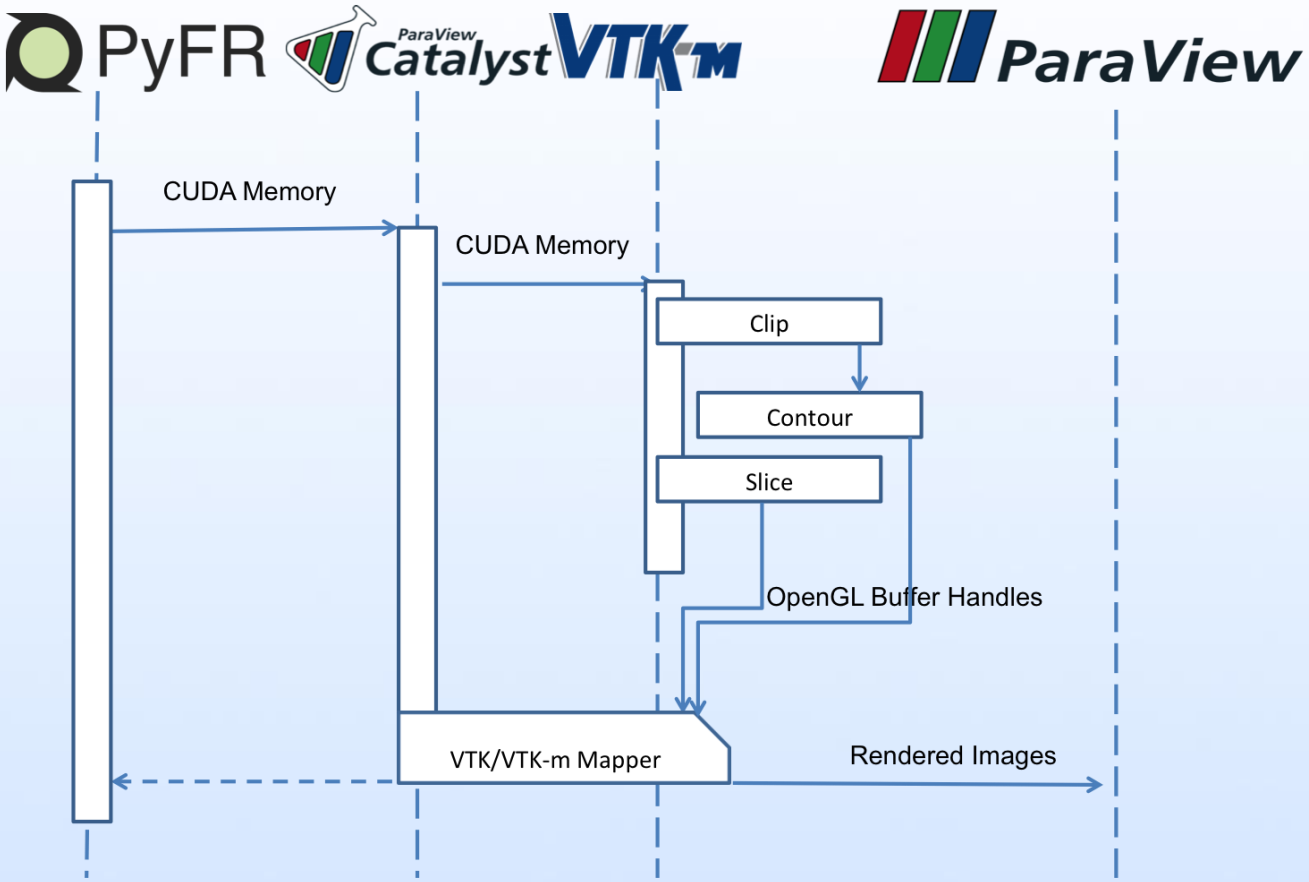
- Bhushan Desam (bdesam)
- Tom Fogal (tfogal)
- Peter Messmer (pmessmer)
- Jeremy Purches (jpurches)

The VTK-m / Catalyst code was written by:

- Utkarsh Ayachit
- T.J. Corona
- Robert Maynard
- Tom Fogal

Note: Thank You Don Maxwell and Jack Wells and for OLCF Allocation.

Demonstration

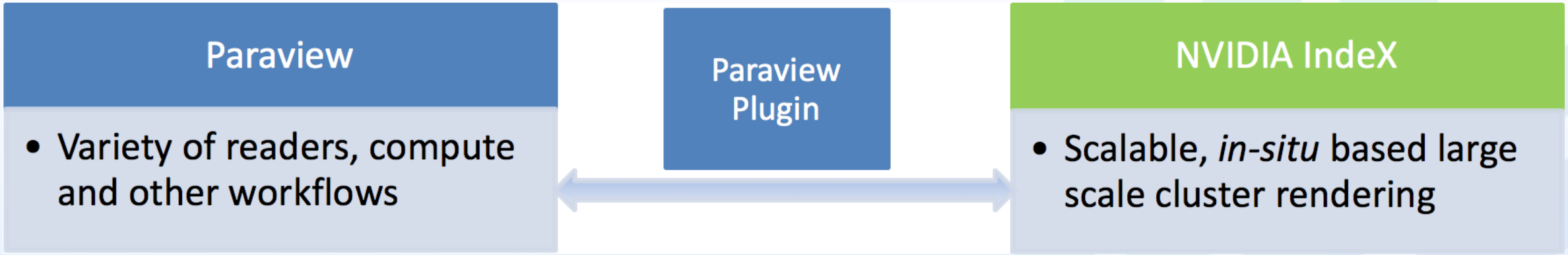


NVIDIA IndeX Integration

- Volume rendering is an important analysis tool, but challenges grow with data sizes when interactivity is essential for exploratory science in supercomputing
- NVIDIA IndeX is a scalable volume visualization solution that provides interactive performance on GPU clusters and supercomputers
- ParaView users can now access NVIDIA IndeX commercial solution without changing existing workflows
- Early version with support for structured grids and time-series data will be available for evaluation in Q1 2016

Volume visualization of a terabyte size tornado dataset in ParaView with NVIDIA IndeX (Dataset courtesy of Leigh Orf, U. of Wisconsin-Madison and Rob Sisneros, NCSA)

Workings of NVIDIA IndeX for ParaView



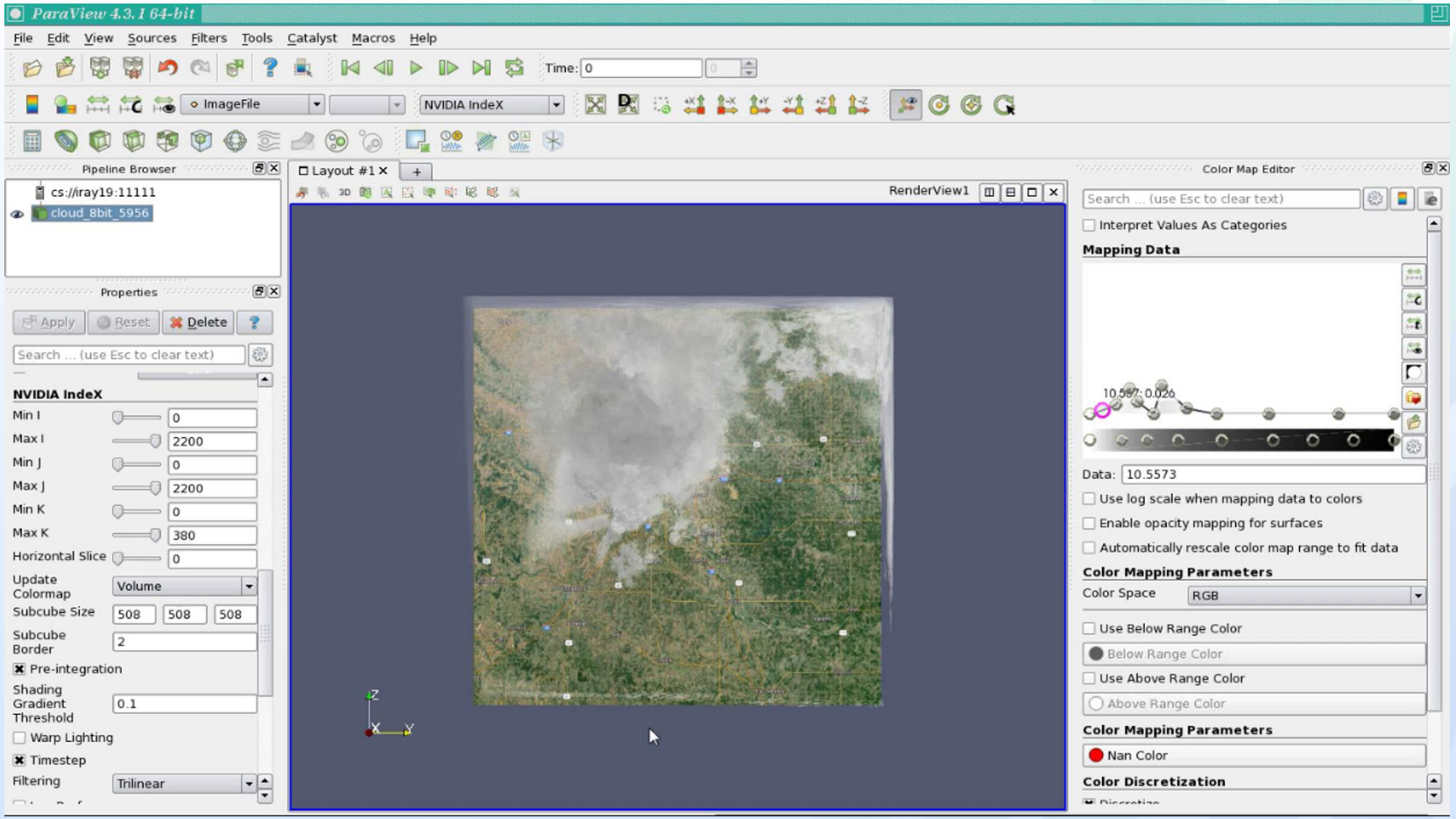
- Simple deployment on the user side
  - Load as any other Paraview plugin
- Plugin adds a new “Representation”
  - Switch to “NVIDIA IndeX” Representation in the GUI
- Cluster rendering in Paraview’s client-server mode via MPI

Acknowledgements

Kitware, Inc @kitware.com

- Utkarsh Ayachit (utkarsh.ayachit)
- Berk Geveci (berk.geveci)

NVIDIA @nvidia.com

- Mahendra Roopa (mroopa)
- Bhushan Desam (bdesam)
- Tom-Michael Thamm (tthamm)

Demonstrations