Large-scale to Exascale Data Exploration and Visualization with ParaView

2025-03-25 François Mazen

Contact: francois.mazen@kitware.com

Who am I?

- François Mazen

- Director of Scientific Visualization
- Kitware Europe, Lyon, France

- Open Source Software Enthusiast

Kitware / Leader in AI & scientific open source solutions

Software development

Based on open source tools
300+ active projects worldwide

Sustained Growth

Since creation of the company
100% employee-owned

230 employees Worldwide

6 offices across USA/Europe

65% staff with PhD or Master

High Level customer expertise

20+ years of expertise

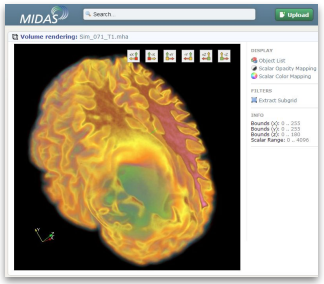
Kitware USA, 1998
Kitware Europe, 2010

Revenue 2020

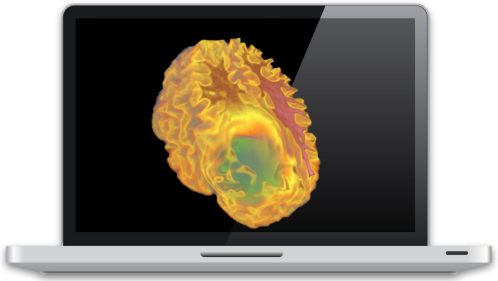
$39M consolidated

kitware

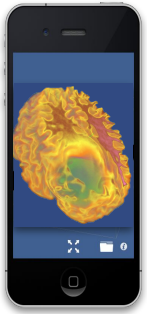
Applications / Universal Platforms



Web



Desktop



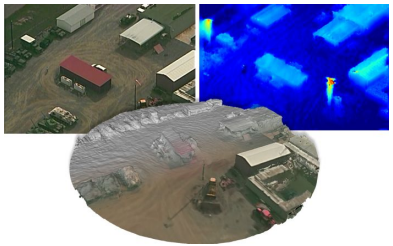
Mobile

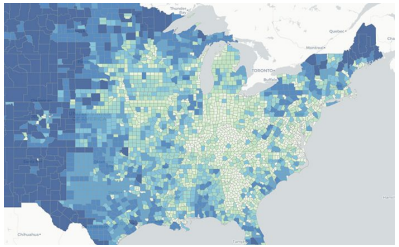
Cloud /HPC

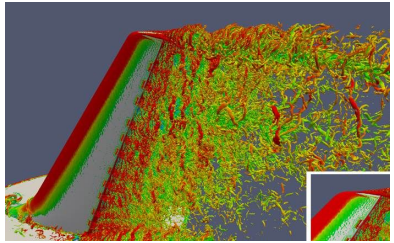
kitware
Platforms

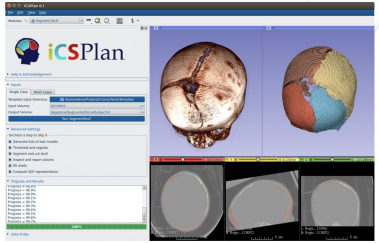
3D Slicer
ParaView
KWIVER
mstk
VTK
Pulse Physiology Engine
CMake
Resonant
tomviz
ITK

Areas of expertise / Built on open source









Computer Vision

Data and Analytics

Scientific Computing

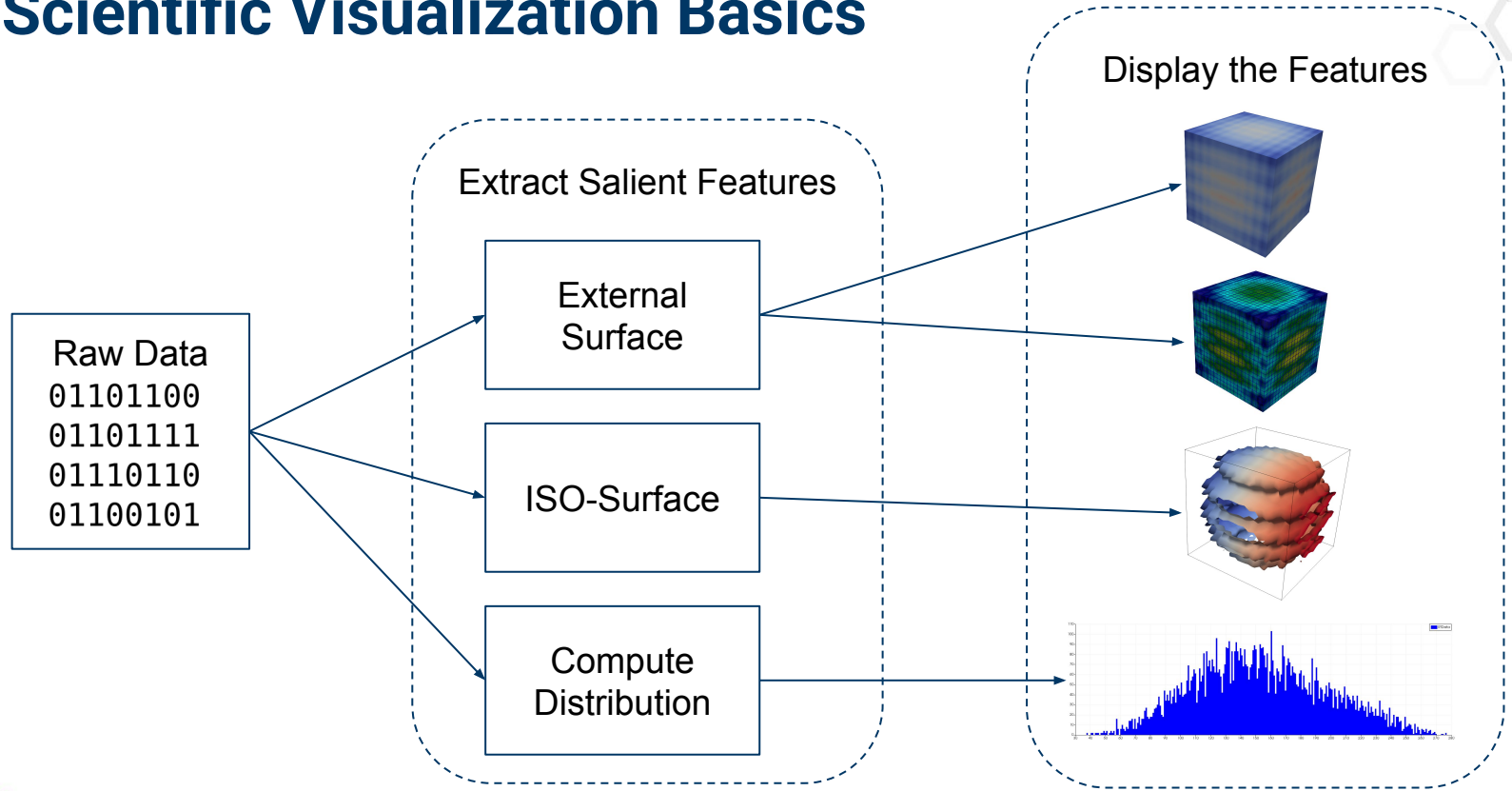
Medical Computing

Software Solutions

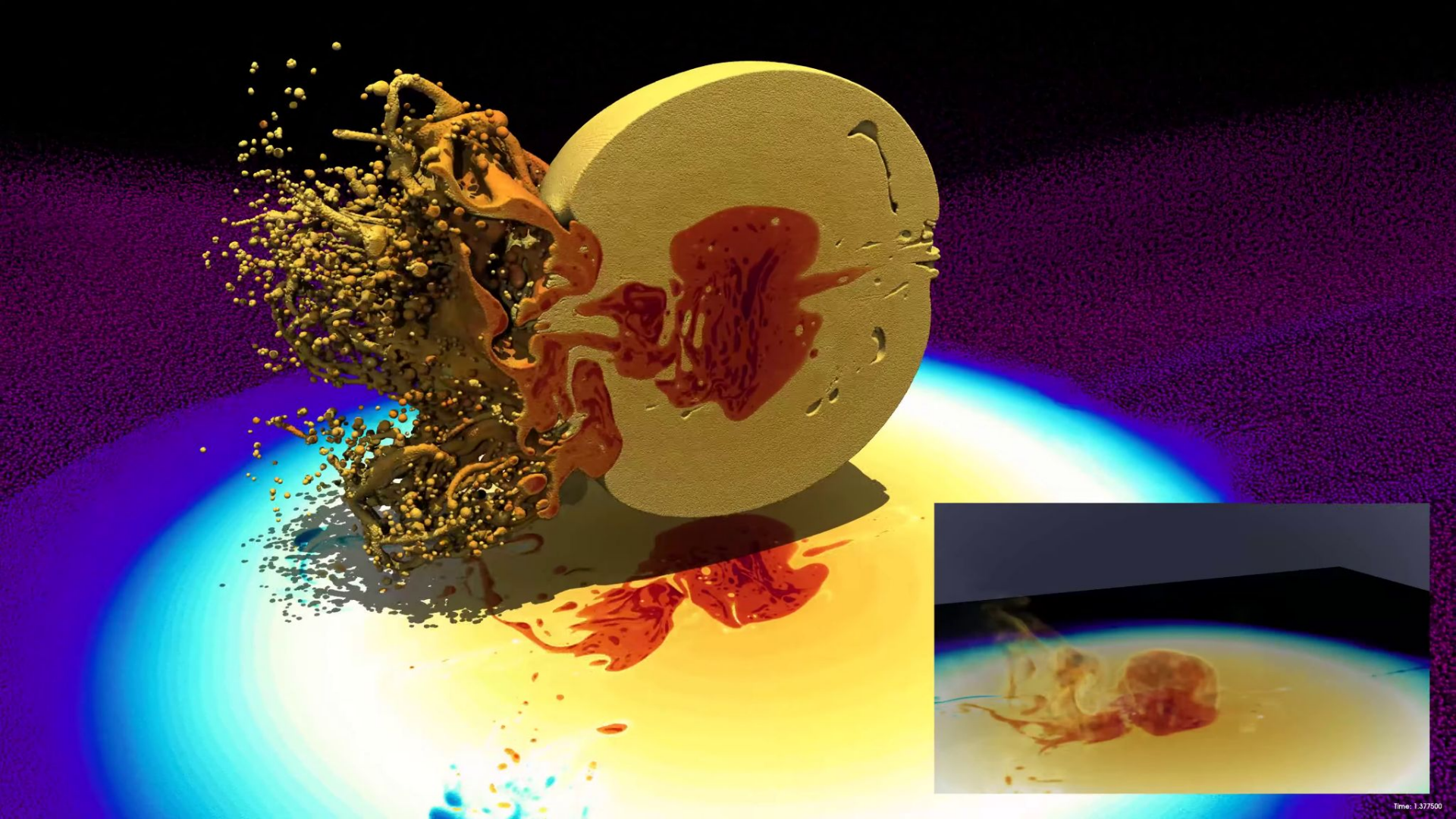
kitware

Kitware / Services

Scientific Visualization Basics



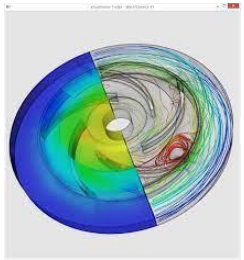
The birth of Jupiter: a first visualization computed on CSCS's 'Alps' machine

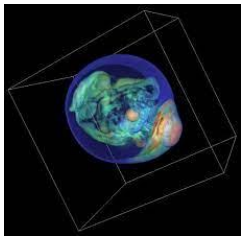

Time: 1.377500

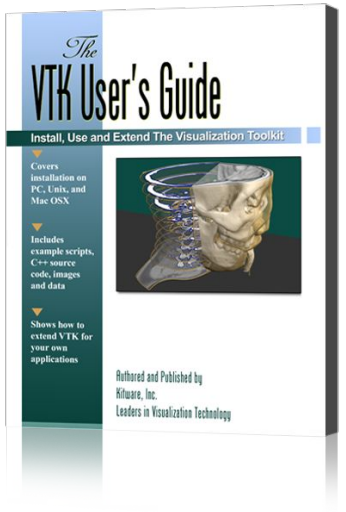
ParaView

VTK / Cross-Platform Visualization Toolkit (1993)

- **Open-source (BSD-3 licence), freely available, cross-platform toolkit for post-processing and visualization of scientific data**

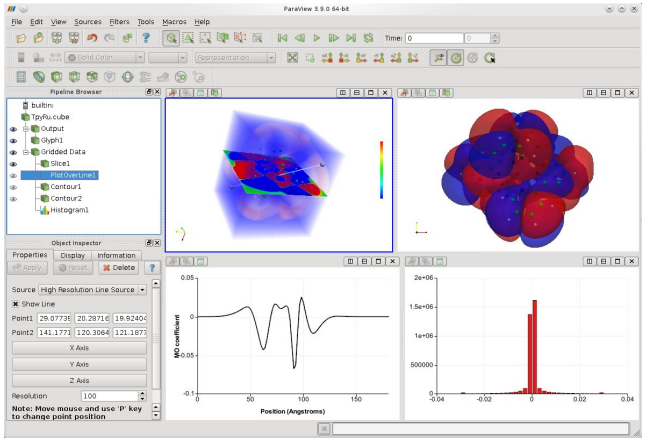


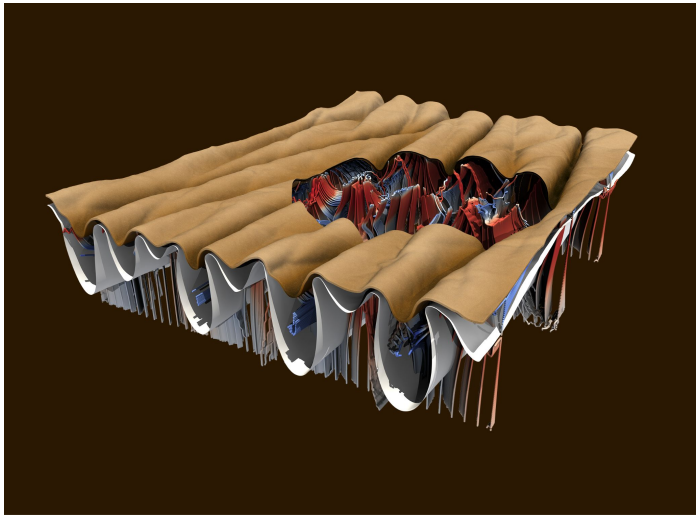




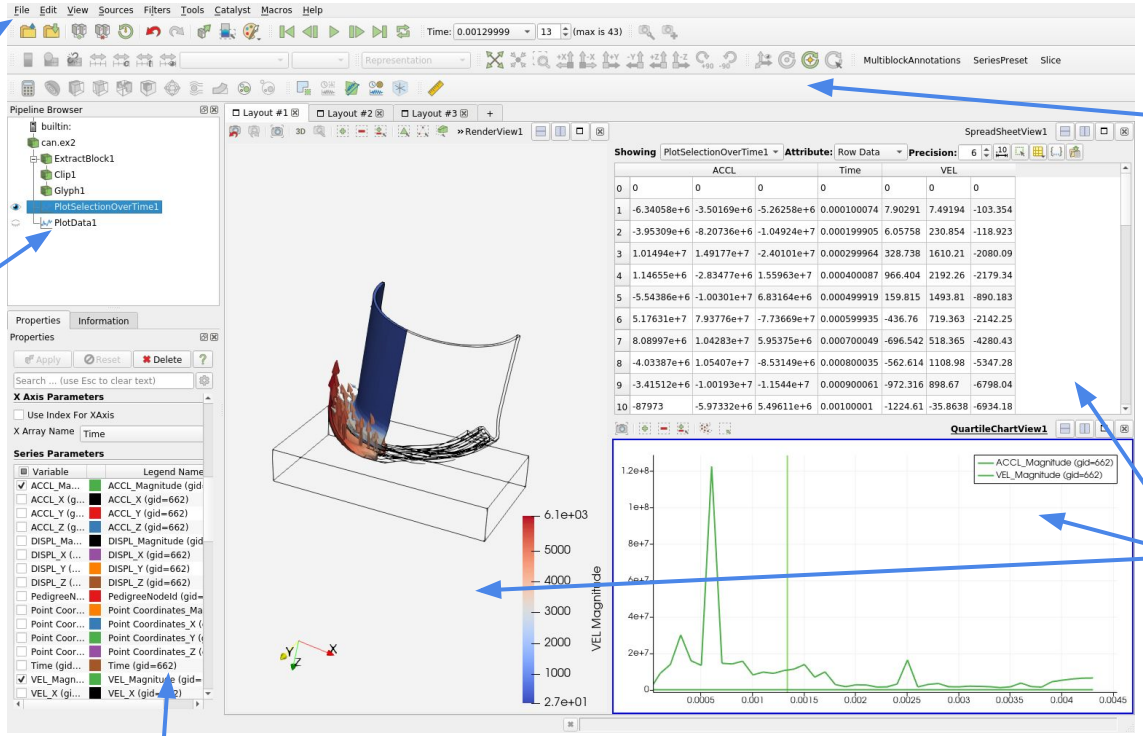
ParaView / High-Performance Post-Processing (2002)

- **Open-source, multi-platform, data analysis and visualization application**
- **Analysis of extremely large datasets using distributed memory computing resources**





ParaView - Graphical User Interface



Menu bar

Pipeline Browser

Toolbars

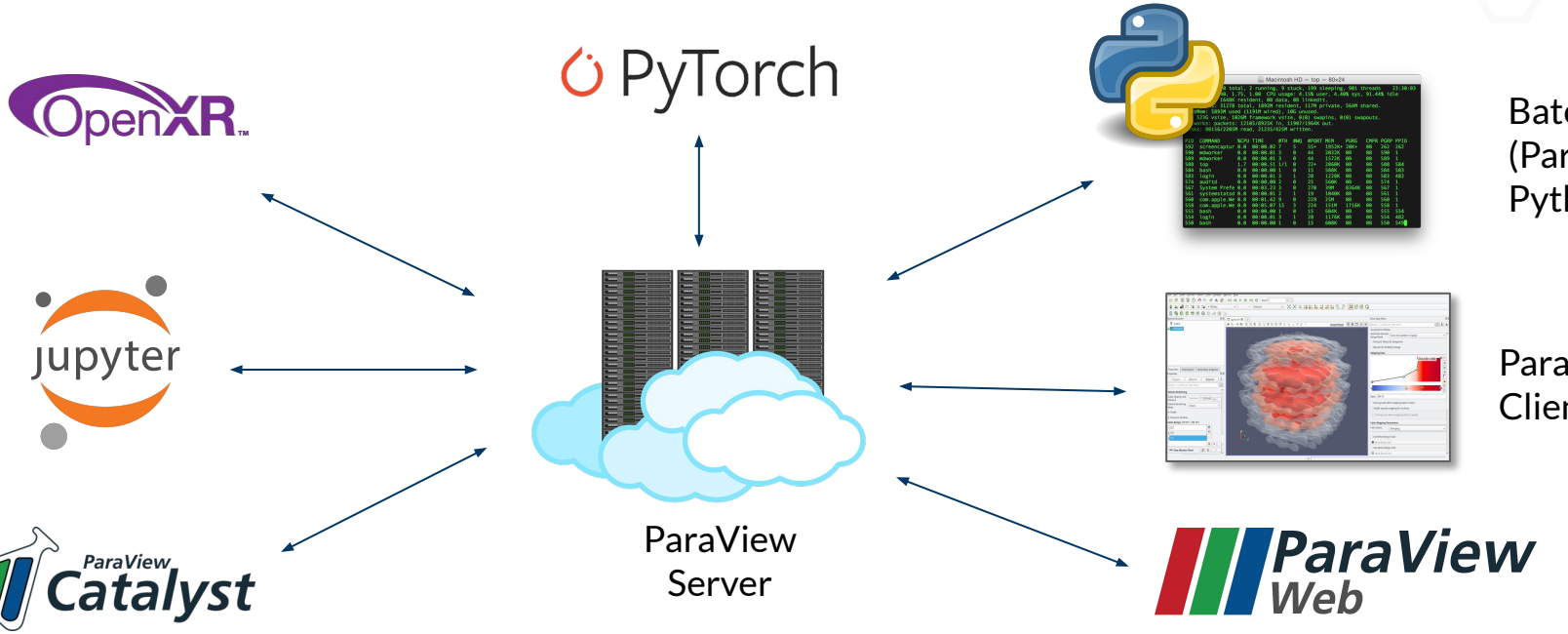
View(s)

Properties panel

ParaView Community

- **Open Source Software (BSD license)**
- **Run on most of Top500 HPCs**
- **300000+ download yearly from Kitware servers**
  - More users via other unknown download channel (Linux packaging, Enterprise distribution…)
- **157k commits made by 339 contributors since 2000**
- **1.6M lines of code**

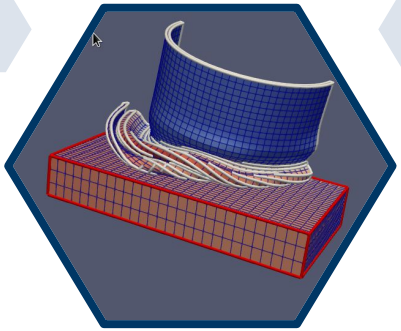
ParaView Ecosystem



kitware

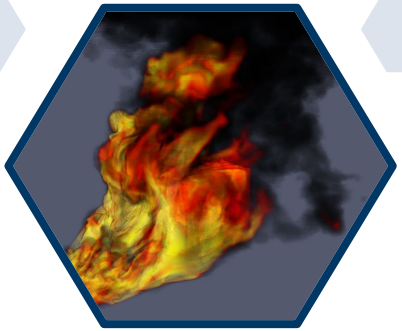
Features / Application Domains

Fluid Dynamic



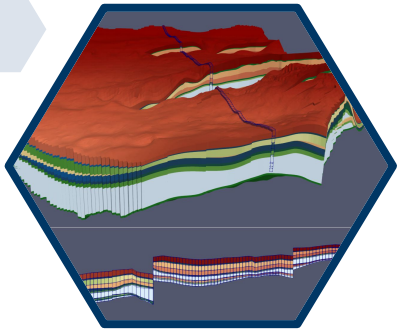
Structural Analysis

Medical



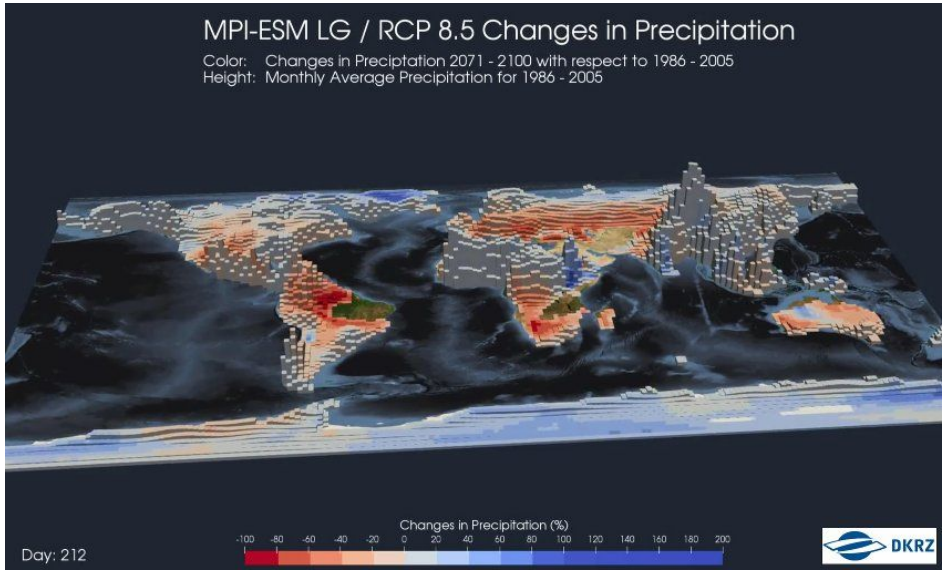
Particles

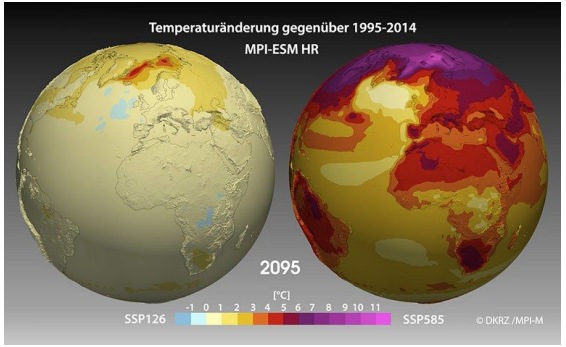
Astrophysic



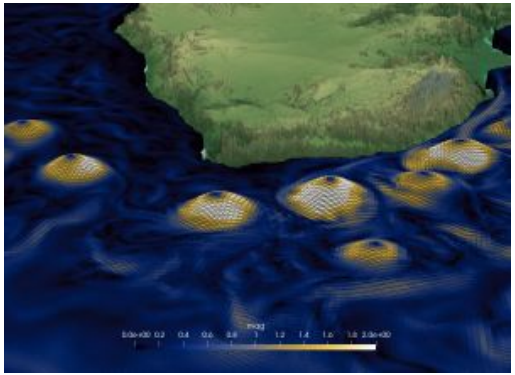
Geoscience

ParaView Use Cases: Climate Simulation (DKRZ)





IPCC Report



https://www.dkrz.de/de/kommunikation/klimasimulationen

Customers & Partners / Nvidia

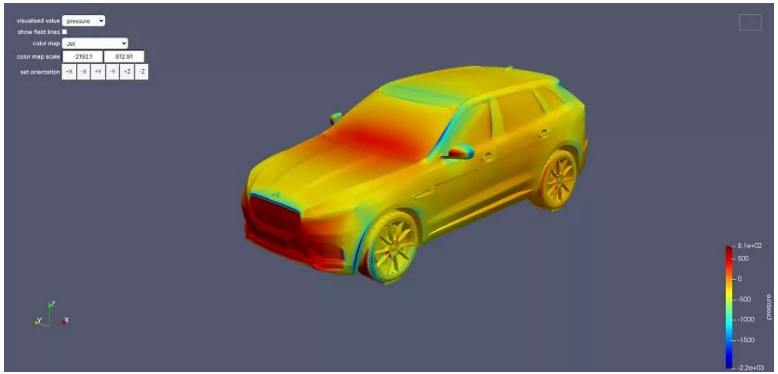
- **On-Demand rendering**
  - Omniverse ParaView Connector
- **3D Volumetric visualization**
  - ParaView plugin for Index
- **Ray-tracing**
  - OptiX integration with ParaView

Customers & Partners / Intel

- **Distributed Rendering on CPUs**
  - The Open, Scalable, and Portable Ray Tracing Engine (OSPRay)
- **Code parallelization**
  - Threading Building Blocks (TBB)
- **Intel GPU support**

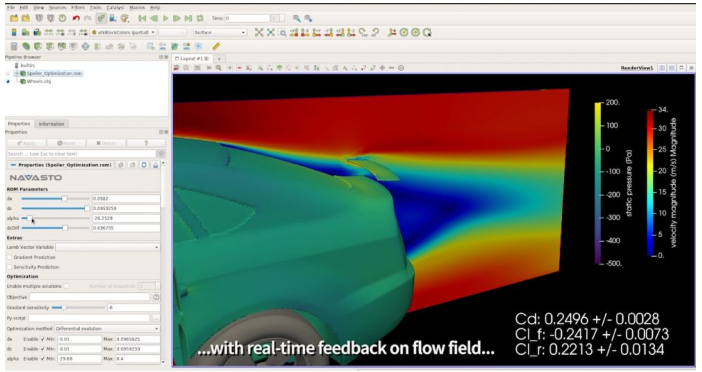
AI in ParaView

Ansys SIMAI



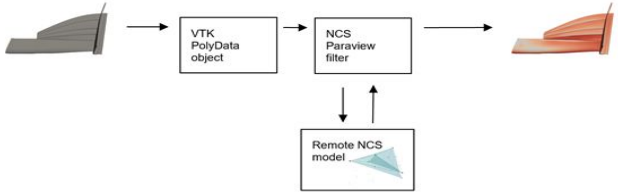
https://www.ansys.com/ai

NAVASTO



https://www.navasto.de/software/navdesign_paraview/

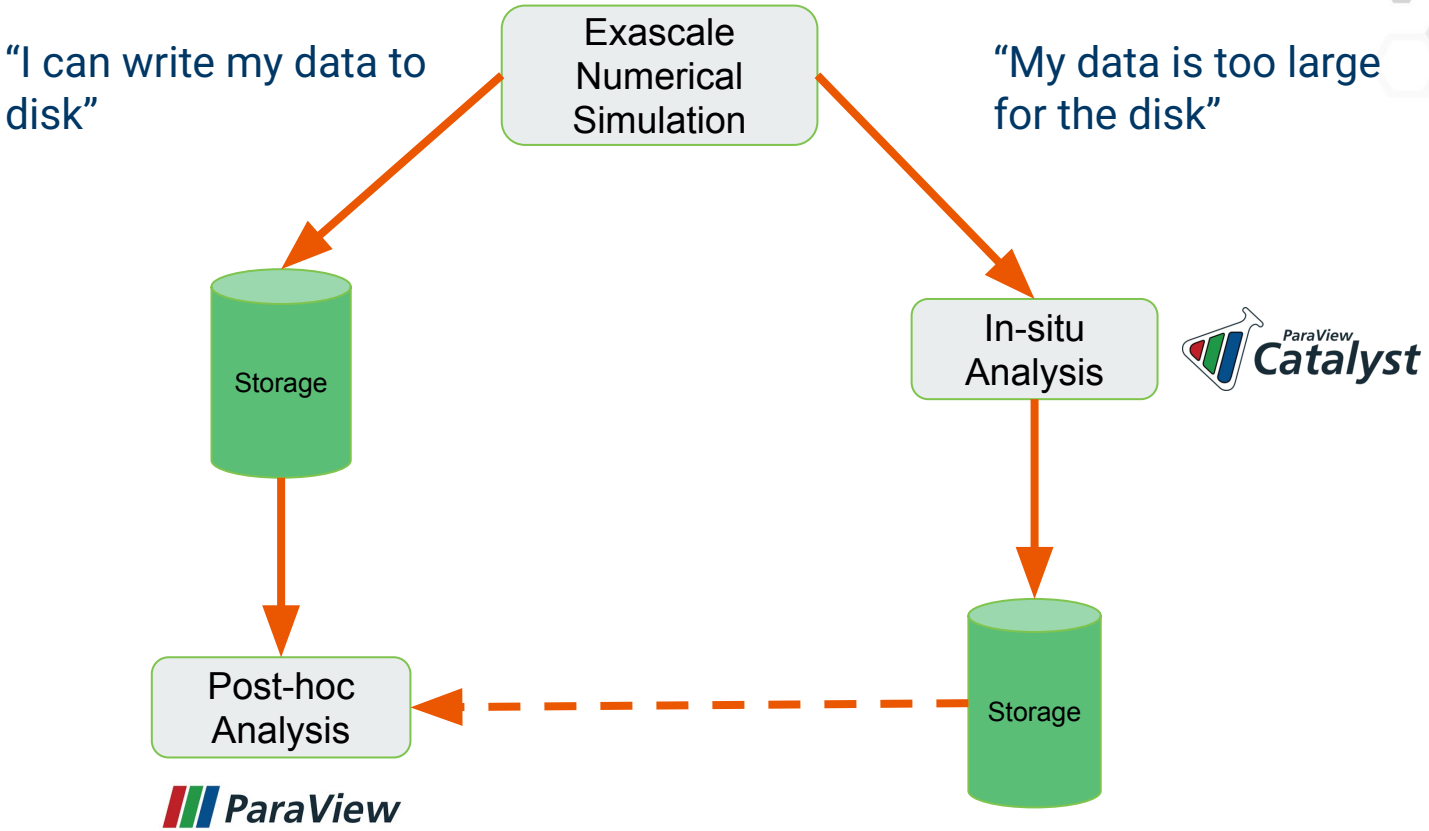
NEURAL CONCEPT



kitware

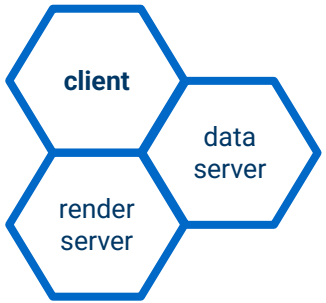
https://www.kitware.com/integrating-geometric-deep-learning-models-into-paraview/

Exascale Challenge #1
Data Size

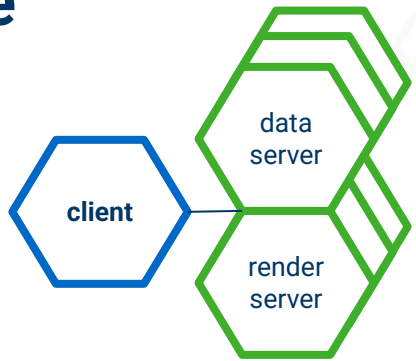


ParaView's Client Server Architecture

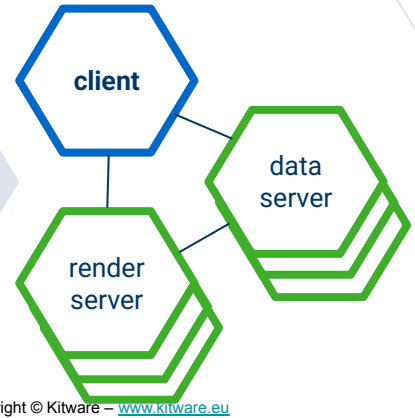
Built-in
paraview



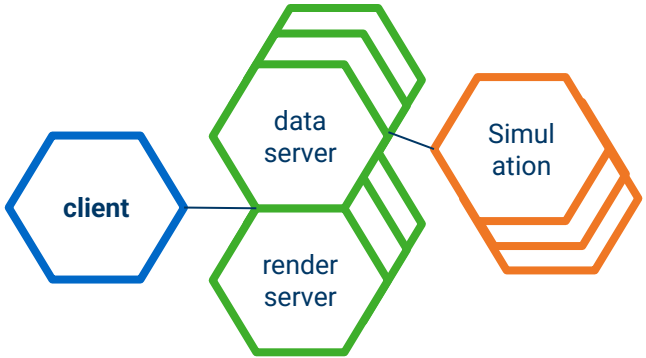
Distributed
pvserver



Graphic Nodes
data/render server



In Situ
catalyst

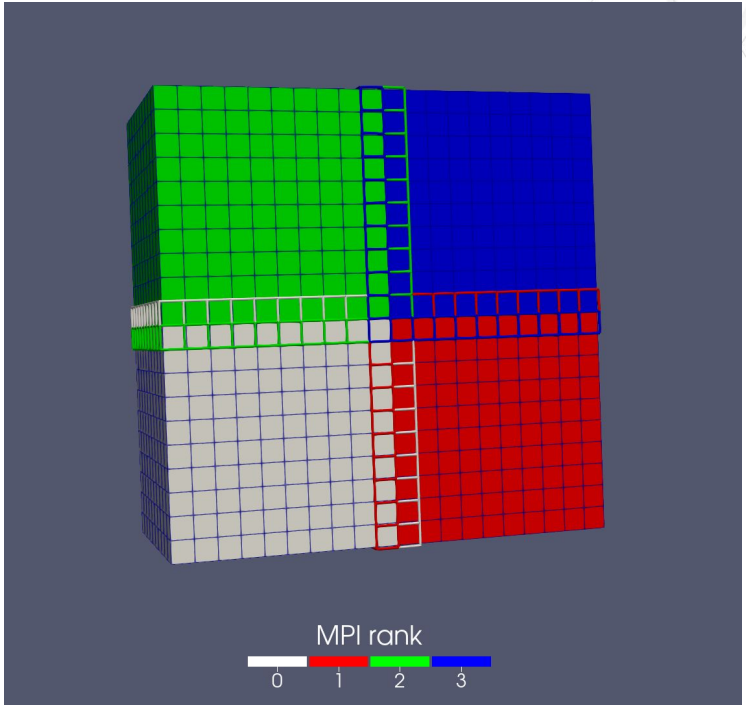


kitware

Data Distribution Analysis

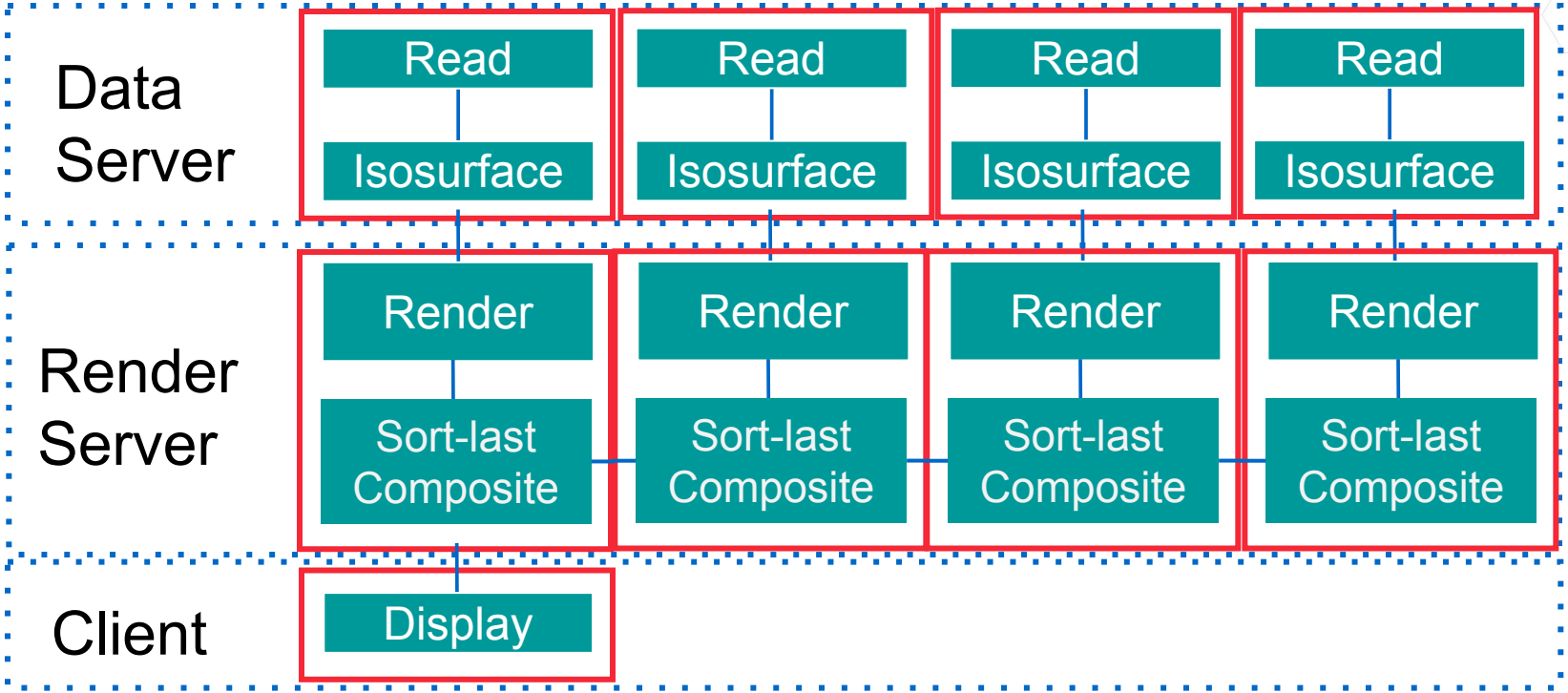
- **Based on MPI standard**
- **Readers distribute data over ranks**
  - load balanced analysis
- **Filters support Ghost Cells**
  - when neighborhood info is needed
- **Filters can redistribute data**
  - ensure load balancing

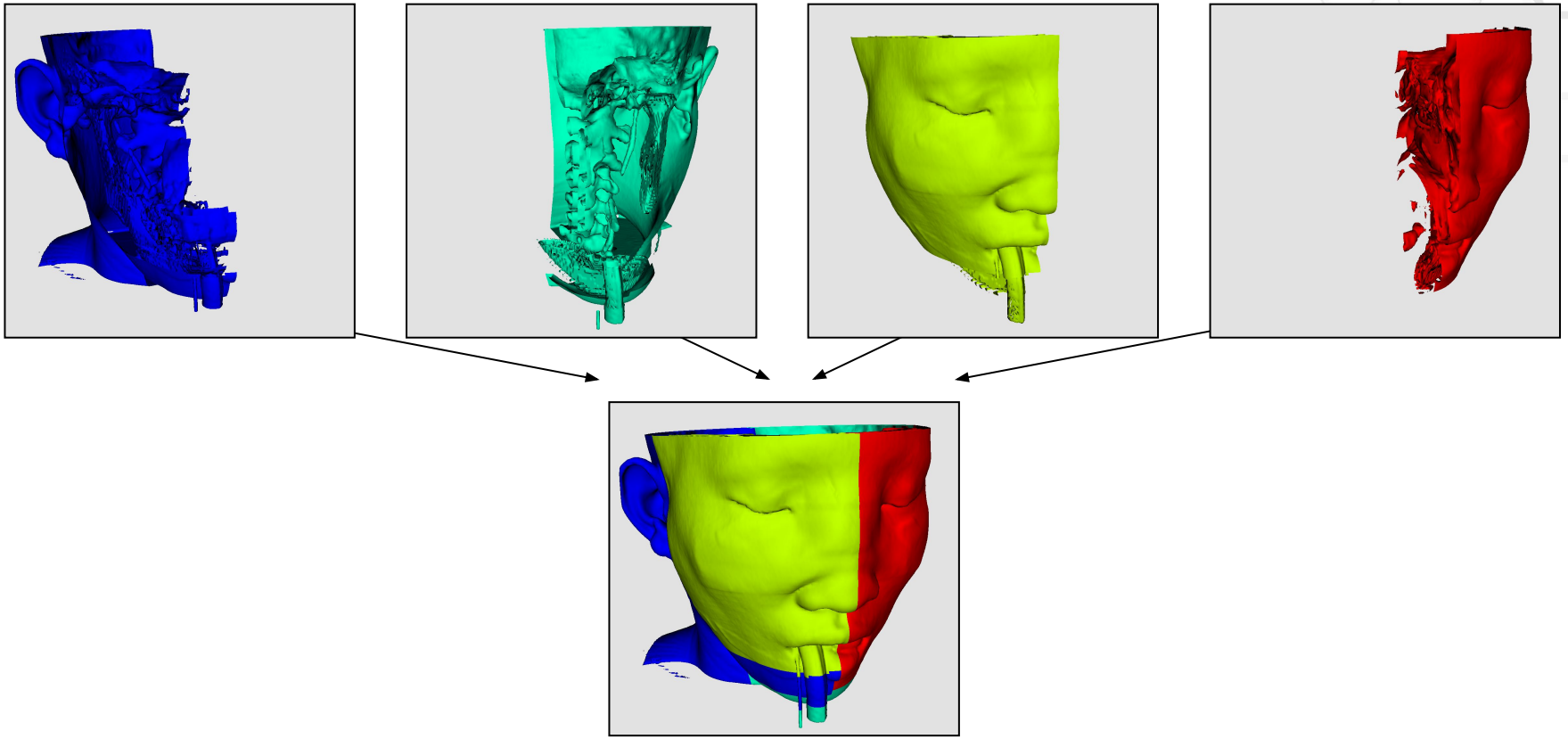
```
$ mpirun -n 4 pvserver
```



Ghost cells in wireframe

ParaView: Distributed Computation

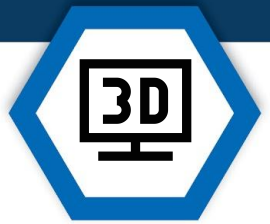




How to connect the client to HPC server?

SSH

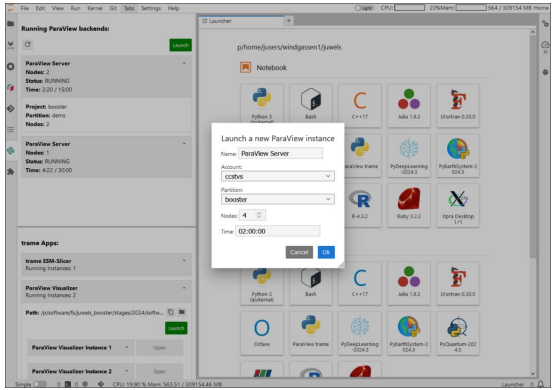
Tunnelling
Reverse Connexion
Double-hop

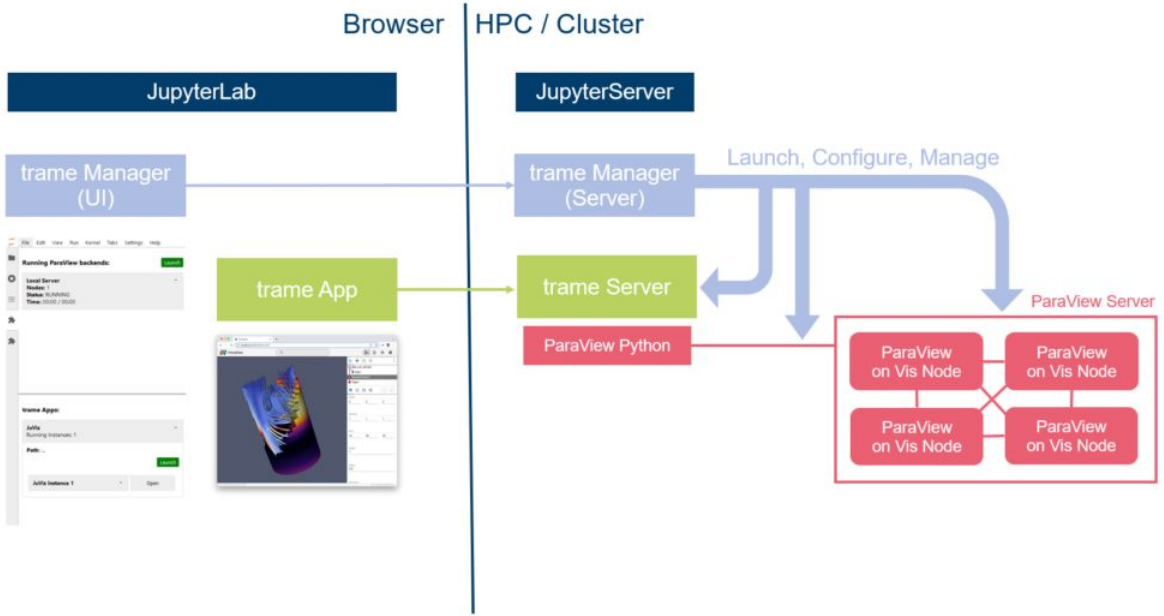


Graphical node
and
Remote desktop

Web
Platform
(trame)

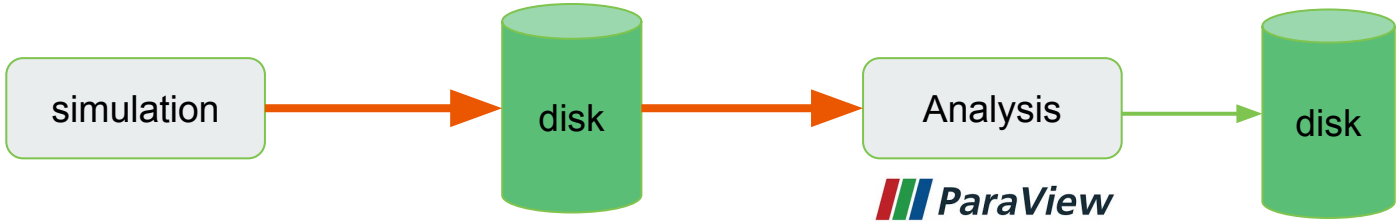
JupyterLab Trame Manager at Jülich Supercomputing Centre

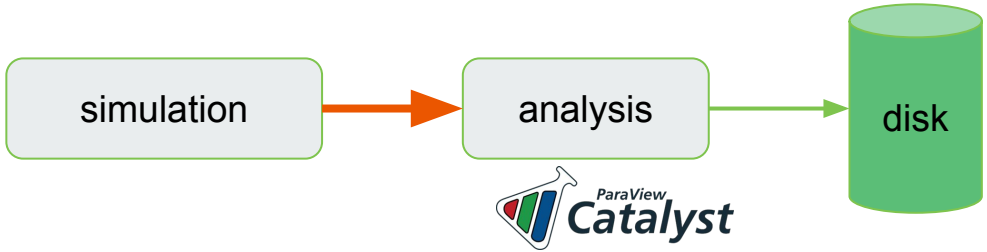




https://www.kitware.com/jupyterlab-trame-manager-at-julich-supercomputing-centre/

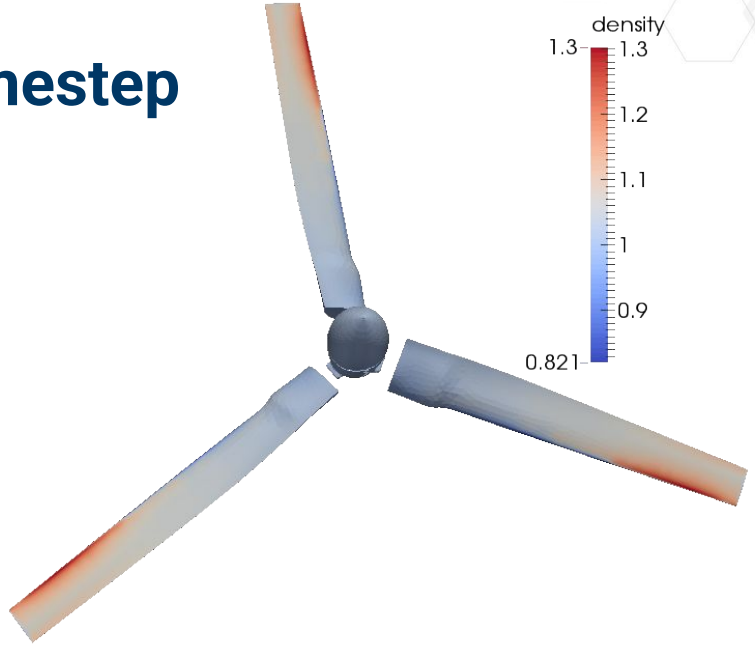
Post-hoc vs In Situ





In-situ Example in numbers

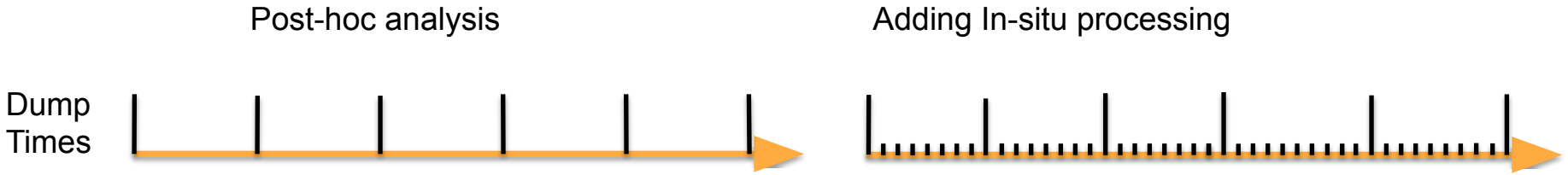
- **Rotorcraft simulation, per timestep**
  - Full data set – 448 MB
  - Surface of blades – 2.8 MB
  - Image – 71 KB



HPCMP CREATE-AV Helios (Army AFDD/AMRDEC) simulation

Access to More Data

- Increase frequency of data saving
- Reduce size of written data



In-situ Tools

- **ADIOS (ADaptable Input Output System)**
- **ASCENT**
  - configuration in YAML
- **libsim**
  - for VisIt
- **Catalyst 1 & 2**
  - for ParaView
- **Conduit**
  - data descriptor, JSON-like
- **Sensei**
  - Interface, XML configuration.

Catalyst 2 Application Programmable Interface (API)

https://catalyst-in-situ.readthedocs.io

Simple, stable C API, independent of ParaView

```
enum catalyst_status catalyst_initialize(const conduit_node* params);
enum catalyst_status catalyst_finalize(const conduit_node* params);
enum catalyst_status catalyst_execute(const conduit_node* params);
enum catalyst_status catalyst_results(conduit_node* params);
enum catalyst_status catalyst_about(conduit_node* params);
```

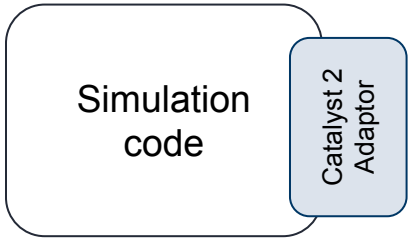
mandatory

optional

Python and Fortran Wrapping available!

Catalyst 2 with ParaView Workflow (1/2)

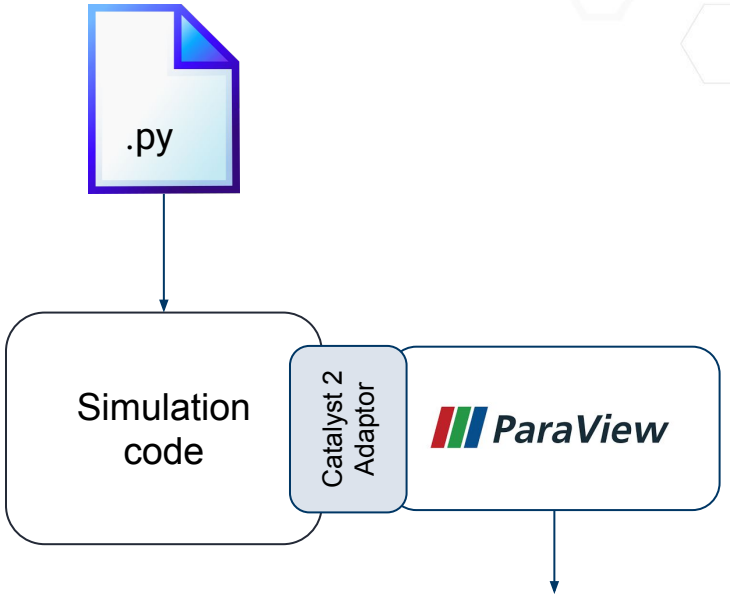
- **Instrument your code with Catalyst 2 once**



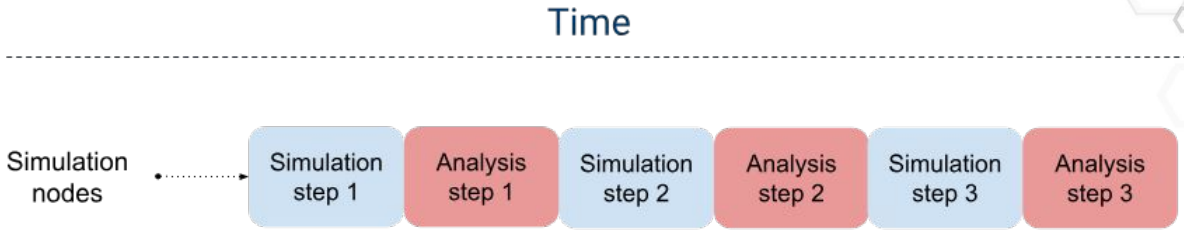
- **Prepare a python pipeline with ParaView**

Catalyst 2 Workflow (2/2)

- **Run your simulation code**
  - Specify the python pipeline in the initialize method
  - Specify the Catalyst 2 implementation (i.e. Catalyst-ParaView)

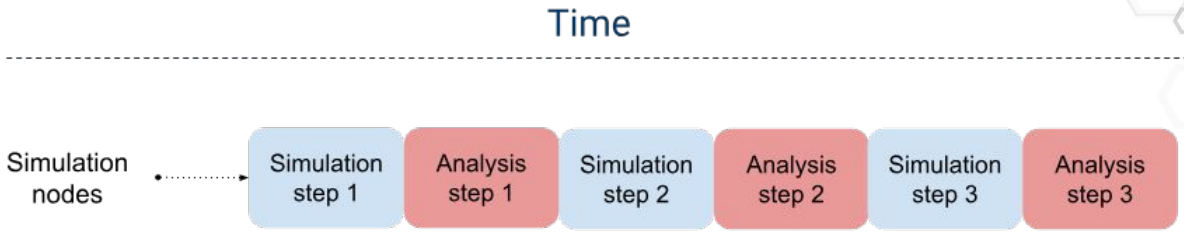


In situ :

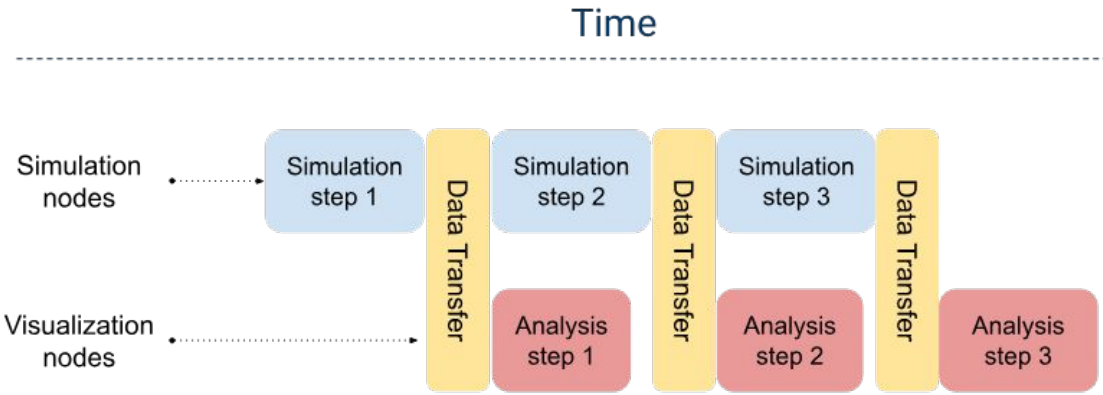


CINECA: Supersonic turbulent boundary layer over a micro-ramp

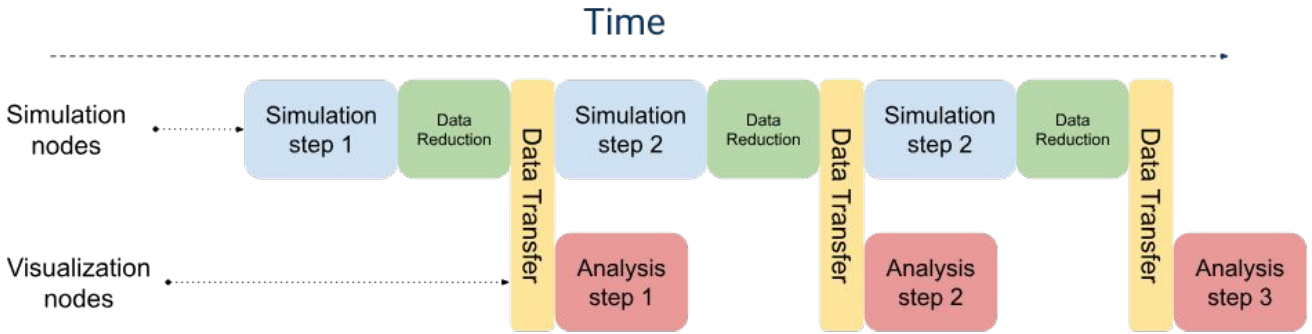
In situ :



In transit :



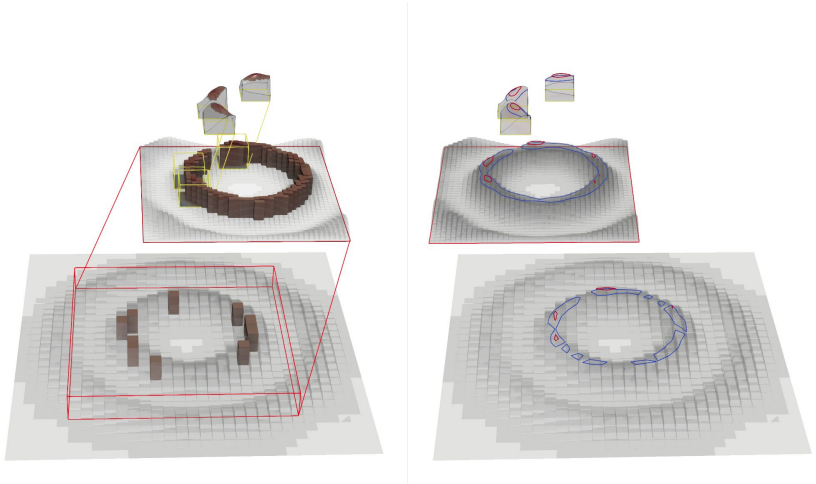
Analyse hybride in situ / in transit



https://www.kitware.com/in-situ-in-transit-hybrid-analysis-using-catalyst-adios2-and-paraview/

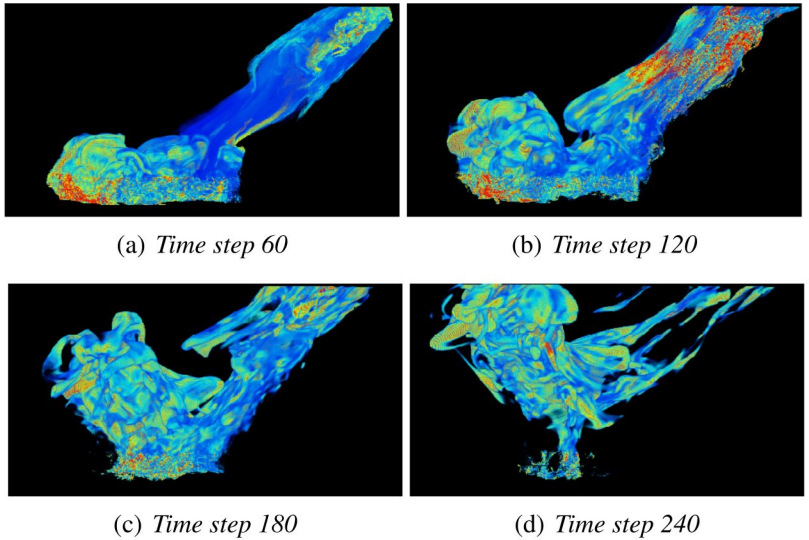
Towards more frugality

Progressive Analysis



Towards a Progressive Open Source Framework for SciVis and InfoVis - IEEEVis PDAV'24

GPU Rendering Out-of-core



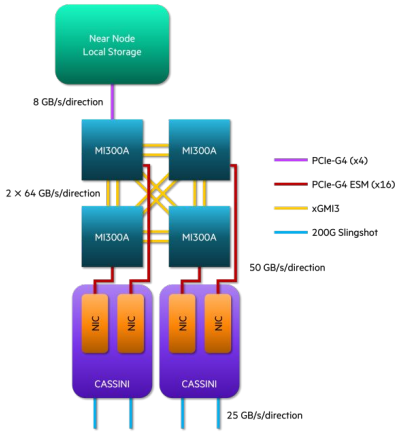
(a) *Time step 60* (b) *Time step 120*

(c) *Time step 180* (d) *Time step 240*

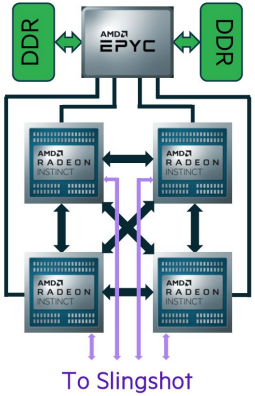
Interactive Visualization of AMR Time Series Data - https://doi.org/10.2312/pgv.20231080

Exascale Challenge #2
Hybrid Architecture

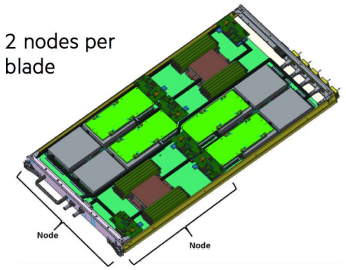
Hybrid CPU / GPU / APU / …PU



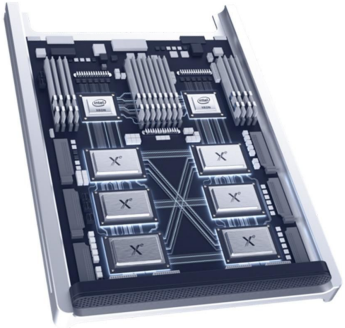
El Capitan (LLNL)



AMD GPU (ORNL)



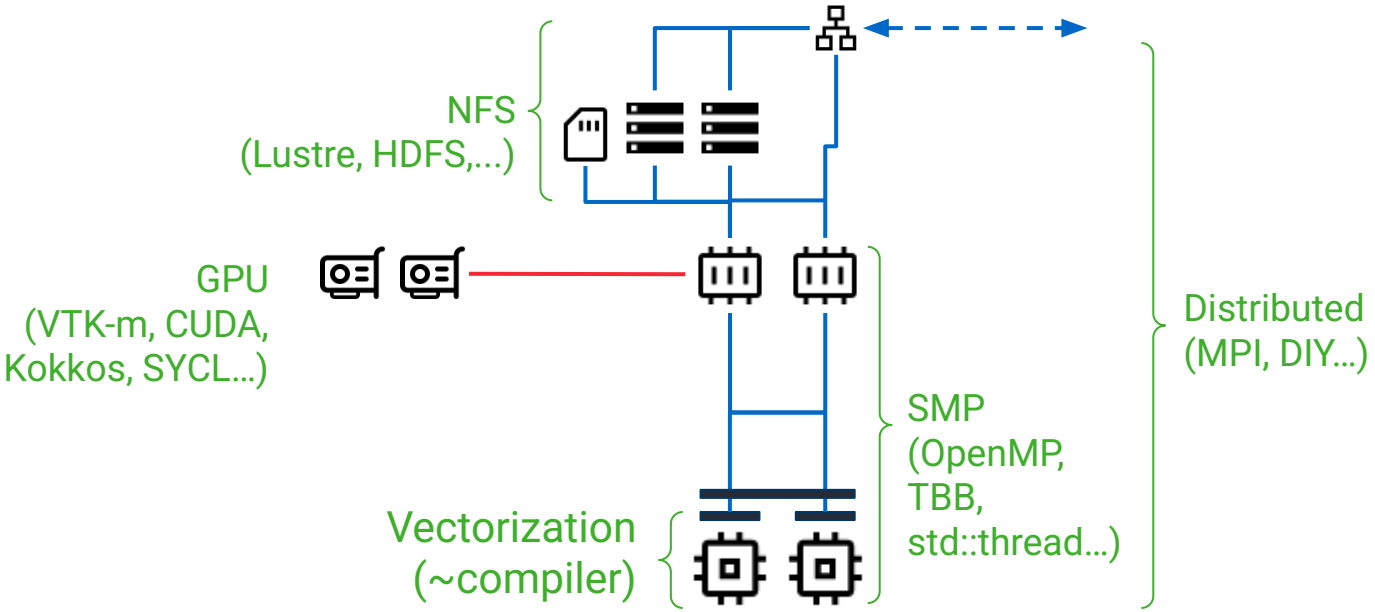
Frontier (ORNL)



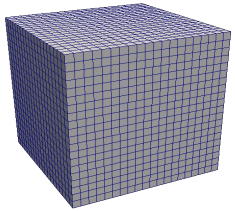
Aurora System Specifications

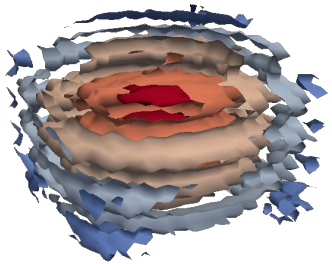
Compute Node
2 Intel Xeon CPU Max Series processors: 64GB HBM on each, 512GB DDR5 each; 6 Intel Data Center GPU Max Series, 128GB HBM on each, RAMBO cache on each; Unified Memory Architecture; 8 SlingShot 11 fabric endpoints

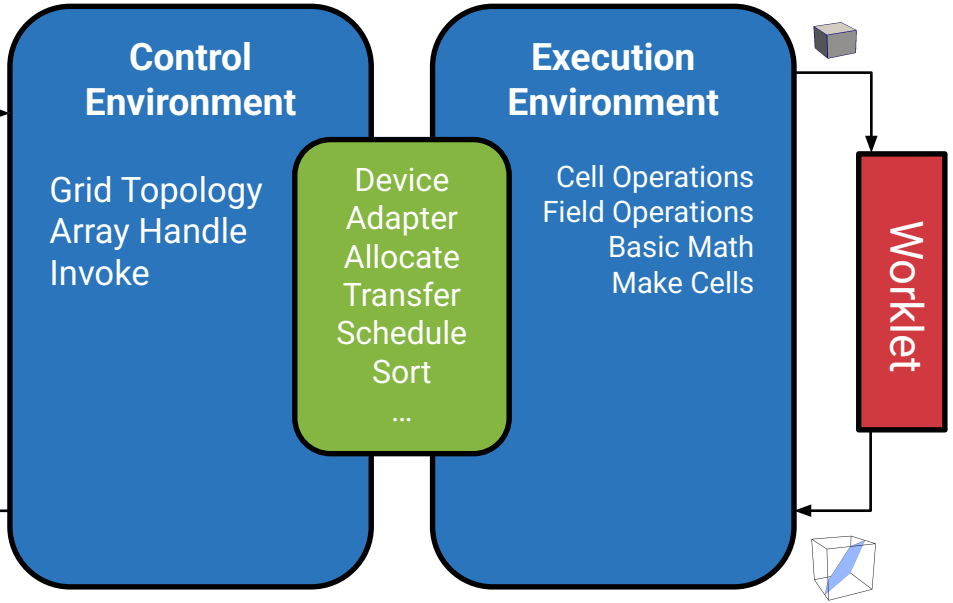
Hybrid Parallel Programming



Viskores / VTK-m







NVIDIA CUDA

kokkos

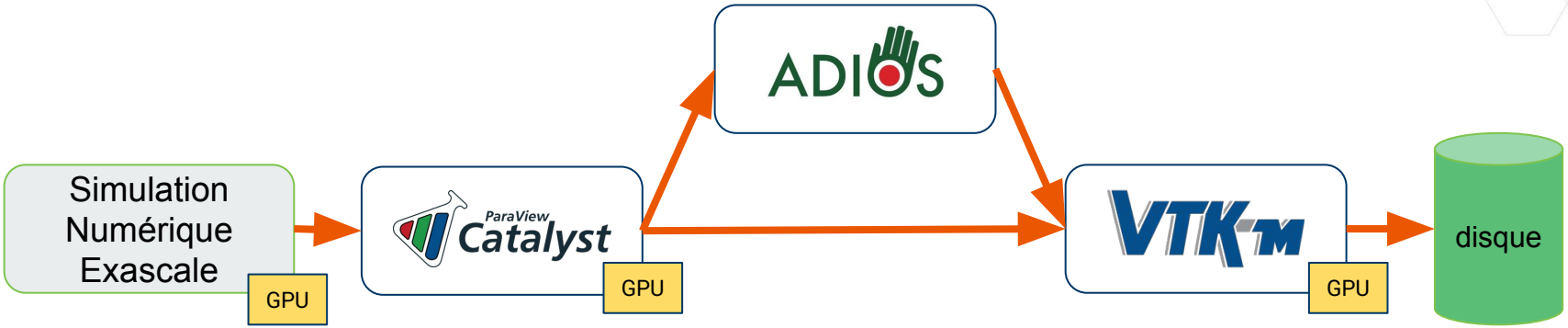
MPI

OpenMP

Intel oneTBB

kitware

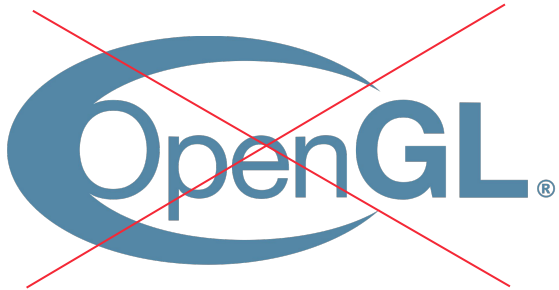
Catalyst GPU-resident workflows



https://www.kitware.com/catalyst2-gpu-resident-workflows/

Exascale Challenge #3
Graphic API

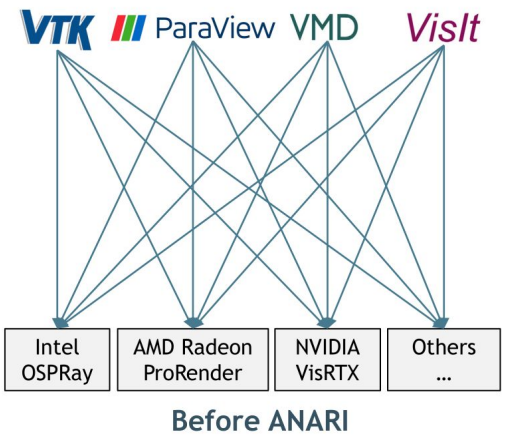
HPC Graphic API “Crisis”

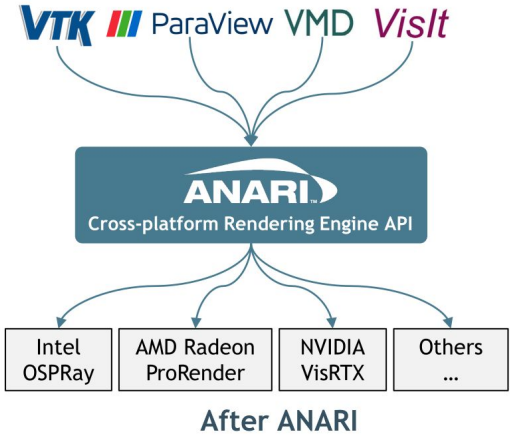


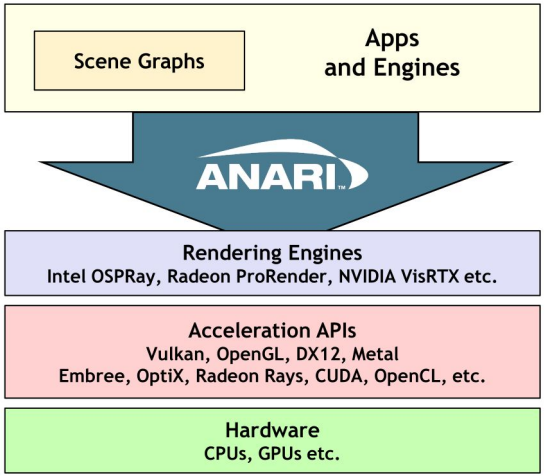
end-of-life 2014

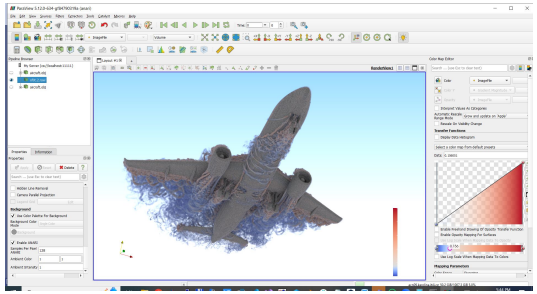
- **HPC GPU without rasterization!**
- **Must use vendor libraries:**
  - Radeon ProRender
  - NVidia VisRTX
  - Intel OSPRay
  - …
- **Or OpenGL emulation on CPU (Mesa/OSMesa)...**

Cross-Platform 3D Rendering Engine API

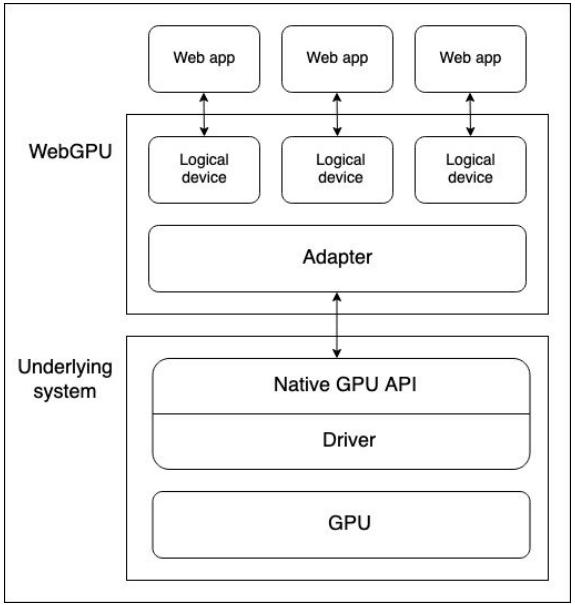








WebGPU



Achieving Interactivity with a Point Cloud of 2 Billion Points in a Single Workstation

Scientific Visualization Challenges at Exascale Era

Data Size

ParaView

ParaView Catalyst

Hybrid Architectures

VTK-m

Graphic API

ANARI™

WebGPU

RFC: Request For Collaboration!

Thank You!

Kitware Europe
kitware@kitware.eu
+33 437-450-415