Hartree Centre
Science & Technology Facilities Council

Benefits of Leveraging Software Defined Visualization (OSPRay)

Sergi Siso
IPCC @ Hartree Centre, STFC Daresbury Laboratory
United Kingdom

PRACE

intel®

Hartree Centre
Science & Technology Facilities Council

NCSA
NATIONAL CENTRE FOR SUPERCOMPUTING APPLICATIONS

Index

Section 1

VISUALIZATION CHALLENGES IN HPC

Hartree Centre
Science & Technology Facilities Council

Why we need visualization in HPC

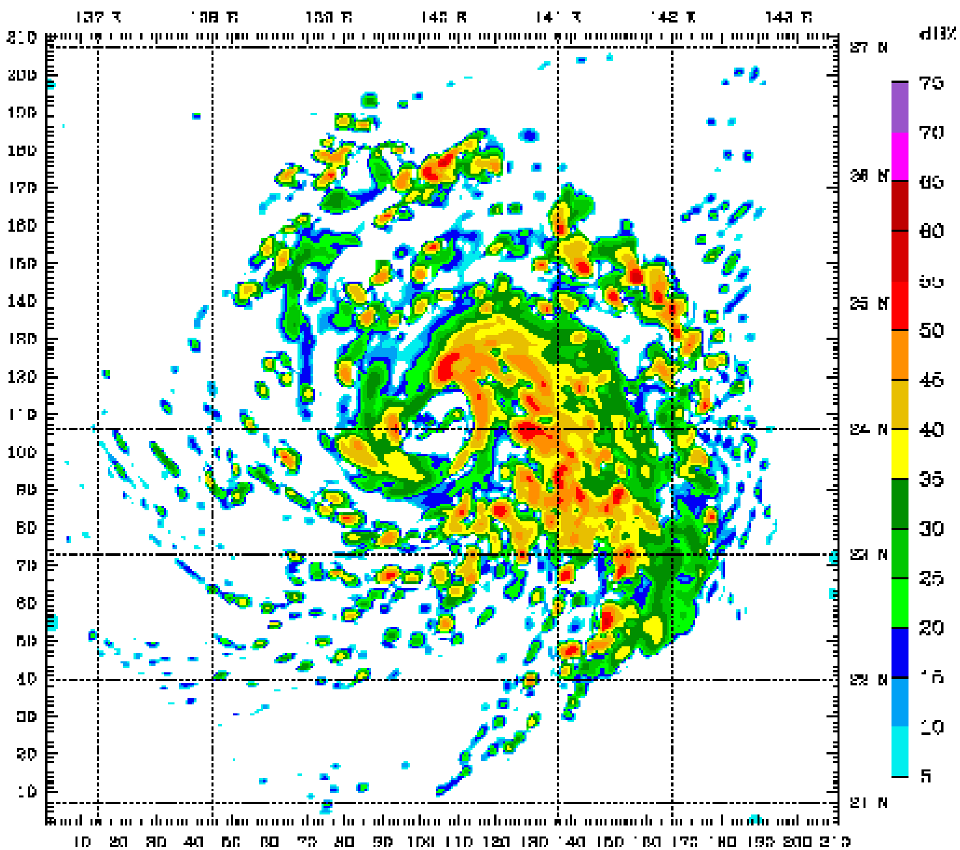
“Yes," said Deep Thought, "I can do it.“

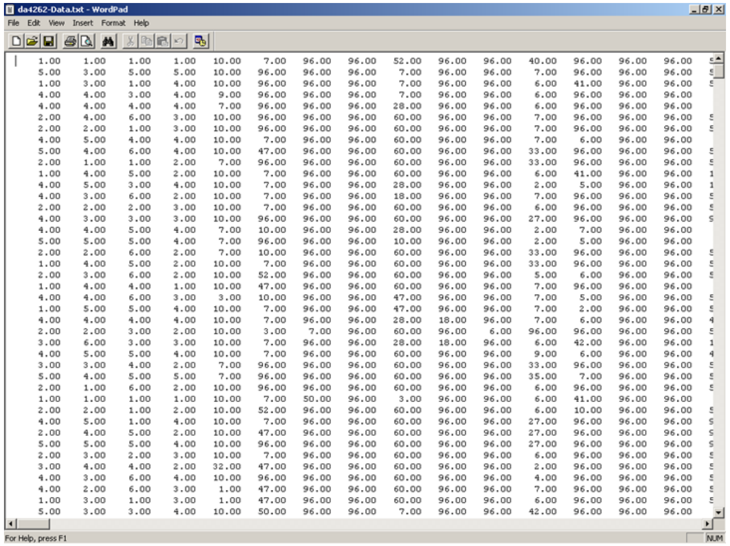
[Seven and a half million years later….]

“The Answer to the Great Question… Of Life, the Universe and Everything… Is… Forty-two,' said Deep Thought, with infinite majesty and calm.”

Douglas Adams, Hitchhiker’s Guide to the Galaxy

Why we need visualization in HPC





1.00	1.00	1.00	1.00	10.00	7.00	96.00	96.00	52.00	96.00	96.00	40.00	96.00	96.00	96.00
5.00	3.00	5.00	5.00	10.00	96.00	96.00	96.00	7.00	96.00	96.00	7.00	96.00	96.00	96.00
1.00	3.00	1.00	4.00	10.00	96.00	96.00	96.00	7.00	96.00	96.00	6.00	41.00	96.00	96.00
4.00	4.00	3.00	4.00	9.00	96.00	96.00	96.00	7.00	96.00	96.00	6.00	96.00	96.00	96.00
4.00	4.00	4.00	4.00	7.00	96.00	96.00	96.00	28.00	96.00	96.00	6.00	96.00	96.00	96.00
2.00	4.00	6.00	3.00	10.00	96.00	96.00	96.00	60.00	96.00	96.00	7.00	96.00	96.00	96.00
2.00	2.00	1.00	3.00	10.00	96.00	96.00	96.00	60.00	96.00	96.00	7.00	96.00	96.00	96.00
4.00	5.00	4.00	4.00	10.00	7.00	96.00	96.00	60.00	96.00	96.00	7.00	6.00	96.00	96.00
5.00	4.00	6.00	4.00	10.00	47.00	96.00	96.00	60.00	96.00	96.00	33.00	96.00	96.00	96.00
2.00	1.00	1.00	2.00	7.00	96.00	96.00	96.00	60.00	96.00	96.00	33.00	96.00	96.00	96.00
1.00	4.00	5.00	2.00	10.00	7.00	96.00	96.00	60.00	96.00	96.00	6.00	41.00	96.00	96.00
4.00	5.00	3.00	4.00	10.00	7.00	96.00	96.00	28.00	96.00	96.00	2.00	5.00	96.00	96.00
4.00	3.00	6.00	2.00	10.00	7.00	96.00	96.00	18.00	96.00	96.00	7.00	96.00	96.00	96.00
2.00	2.00	2.00	3.00	10.00	7.00	96.00	96.00	60.00	96.00	96.00	6.00	96.00	96.00	96.00
4.00	3.00	3.00	3.00	10.00	96.00	96.00	96.00	60.00	96.00	96.00	27.00	96.00	96.00	96.00
4.00	4.00	5.00	4.00	7.00	10.00	96.00	96.00	28.00	96.00	96.00	2.00	7.00	96.00	96.00
5.00	5.00	5.00	4.00	7.00	96.00	96.00	96.00	10.00	96.00	96.00	2.00	5.00	96.00	96.00
2.00	2.00	6.00	2.00	7.00	10.00	96.00	96.00	60.00	96.00	96.00	33.00	96.00	96.00	96.00
1.00	4.00	5.00	2.00	10.00	7.00	96.00	96.00	60.00	96.00	96.00	33.00	96.00	96.00	96.00
2.00	3.00	6.00	2.00	10.00	52.00	96.00	96.00	60.00	96.00	96.00	5.00	6.00	96.00	96.00
1.00	4.00	4.00	1.00	10.00	47.00	96.00	96.00	60.00	96.00	96.00	7.00	96.00	96.00	96.00
4.00	4.00	6.00	3.00	3.00	10.00	96.00	96.00	47.00	96.00	96.00	7.00	5.00	96.00	96.00
1.00	5.00	5.00	4.00	10.00	7.00	96.00	96.00	47.00	96.00	96.00	7.00	2.00	96.00	96.00
4.00	4.00	4.00	4.00	10.00	7.00	96.00	96.00	28.00	18.00	96.00	7.00	6.00	96.00	96.00
2.00	2.00	3.00	2.00	10.00	3.00	7.00	96.00	60.00	96.00	6.00	96.00	96.00	96.00	96.00
3.00	6.00	3.00	3.00	10.00	7.00	96.00	96.00	28.00	18.00	96.00	6.00	42.00	96.00	96.00
4.00	5.00	5.00	4.00	10.00	7.00	96.00	96.00	60.00	96.00	96.00	9.00	6.00	96.00	96.00
3.00	3.00	4.00	2.00	7.00	96.00	96.00	96.00	60.00	96.00	96.00	33.00	96.00	96.00	96.00
5.00	4.00	5.00	5.00	7.00	96.00	96.00	96.00	60.00	96.00	96.00	35.00	7.00	96.00	96.00
2.00	1.00	6.00	2.00	10.00	96.00	96.00	96.00	60.00	96.00	96.00	6.00	96.00	96.00	96.00
1.00	1.00	1.00	1.00	10.00	7.00	50.00	96.00	3.00	96.00	96.00	6.00	41.00	96.00	96.00
2.00	2.00	1.00	2.00	10.00	52.00	96.00	96.00	60.00	96.00	96.00	6.00	10.00	96.00	96.00
4.00	5.00	1.00	4.00	10.00	7.00	96.00	96.00	60.00	96.00	96.00	27.00	96.00	96.00	96.00
2.00	4.00	5.00	2.00	10.00	47.00	96.00	96.00	60.00	96.00	96.00	27.00	96.00	96.00	96.00
5.00	5.00	5.00	4.00	10.00	96.00	96.00	96.00	60.00	96.00	96.00	27.00	96.00	96.00	96.00
2.00	3.00	2.00	3.00	10.00	7.00	96.00	96.00	60.00	96.00	96.00	6.00	96.00	96.00	96.00
3.00	4.00	4.00	2.00	32.00	47.00	96.00	96.00	60.00	96.00	96.00	2.00	96.00	96.00	96.00
4.00	3.00	6.00	4.00	10.00	96.00	96.00	96.00	60.00	96.00	96.00	4.00	96.00	96.00	96.00
4.00	2.00	6.00	3.00	1.00	47.00	96.00	96.00	60.00	96.00	96.00	7.00	96.00	96.00	96.00
1.00	3.00	1.00	3.00	1.00	47.00	96.00	96.00	60.00	96.00	96.00	6.00	96.00	96.00	96.00
5.00	3.00	3.00	4.00	10.00	50.00	96.00	96.00	7.00	96.00	96.00	42.00	96.00	96.00	96.00

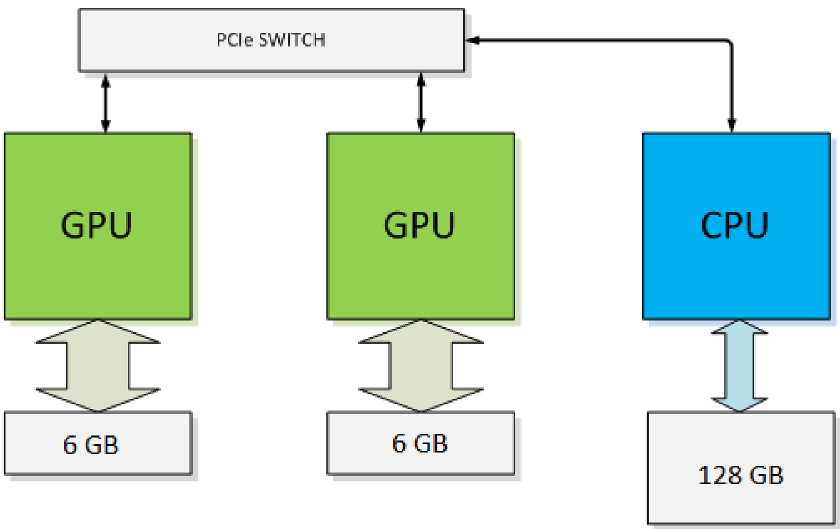
But not just rendering

- In HPC visualization incudes:
  - Feature extraction
  - Isosurfaces / Isovolumes
  - Thresholding
  - Streamlines
  - Clip, Slice, gradient, …
- Visualization is a "big data" problem
  - Data movement problem (IO bottleneck)
  - Needs lots of memory
  - Need scalability to solve

Some HPC visualization workloads are better suited to CPU architectures.

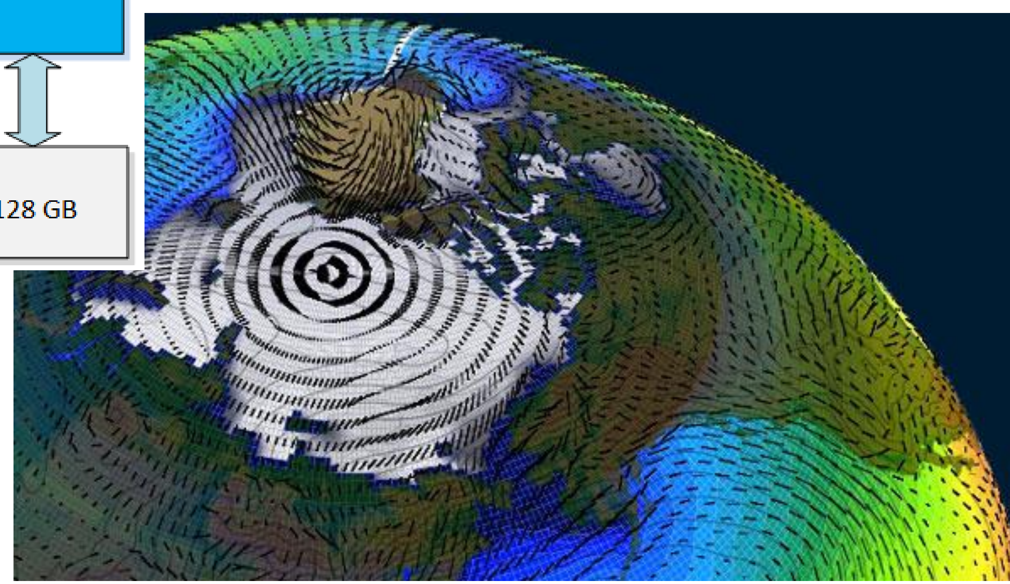
Dataset Size Problem

- Issues to simulate large datasets…



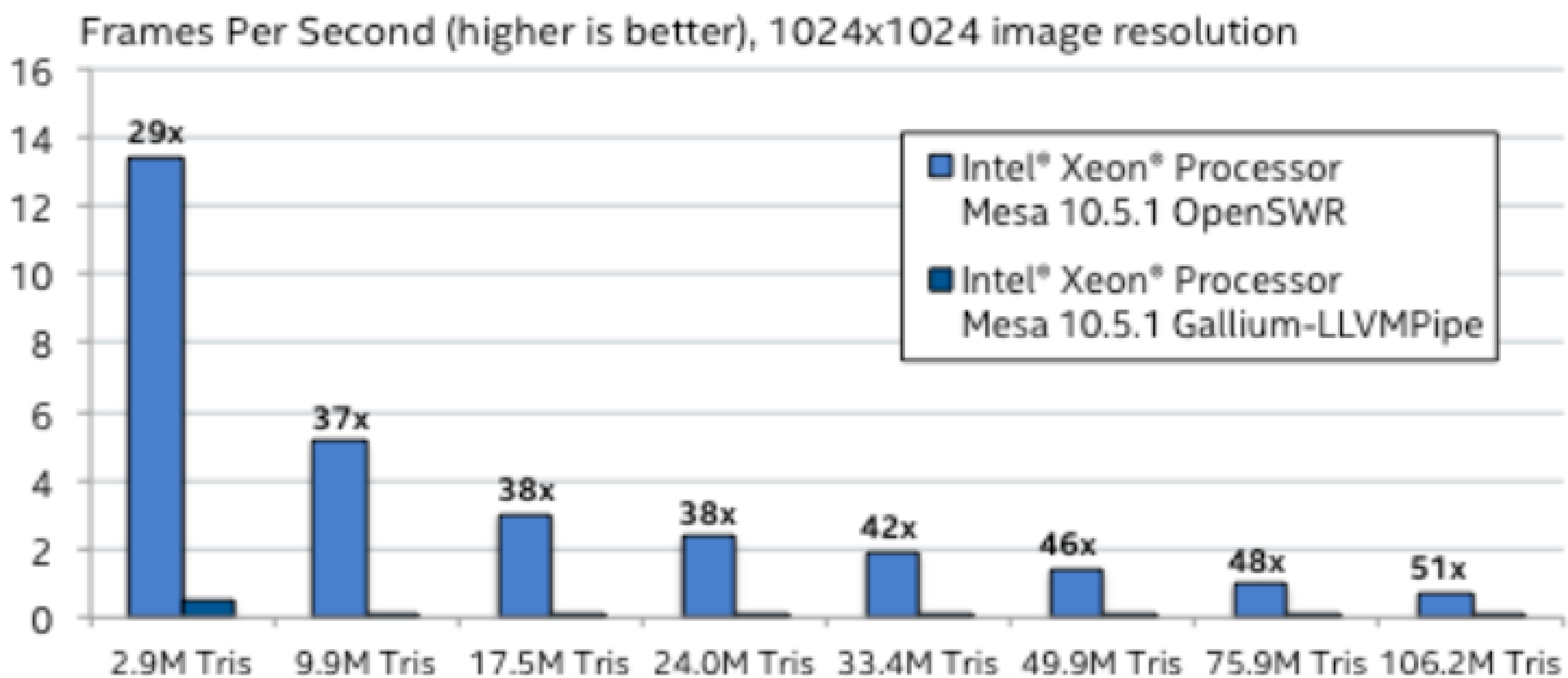
May fit in System Memory but not in GPU memory

PCI bus is slow

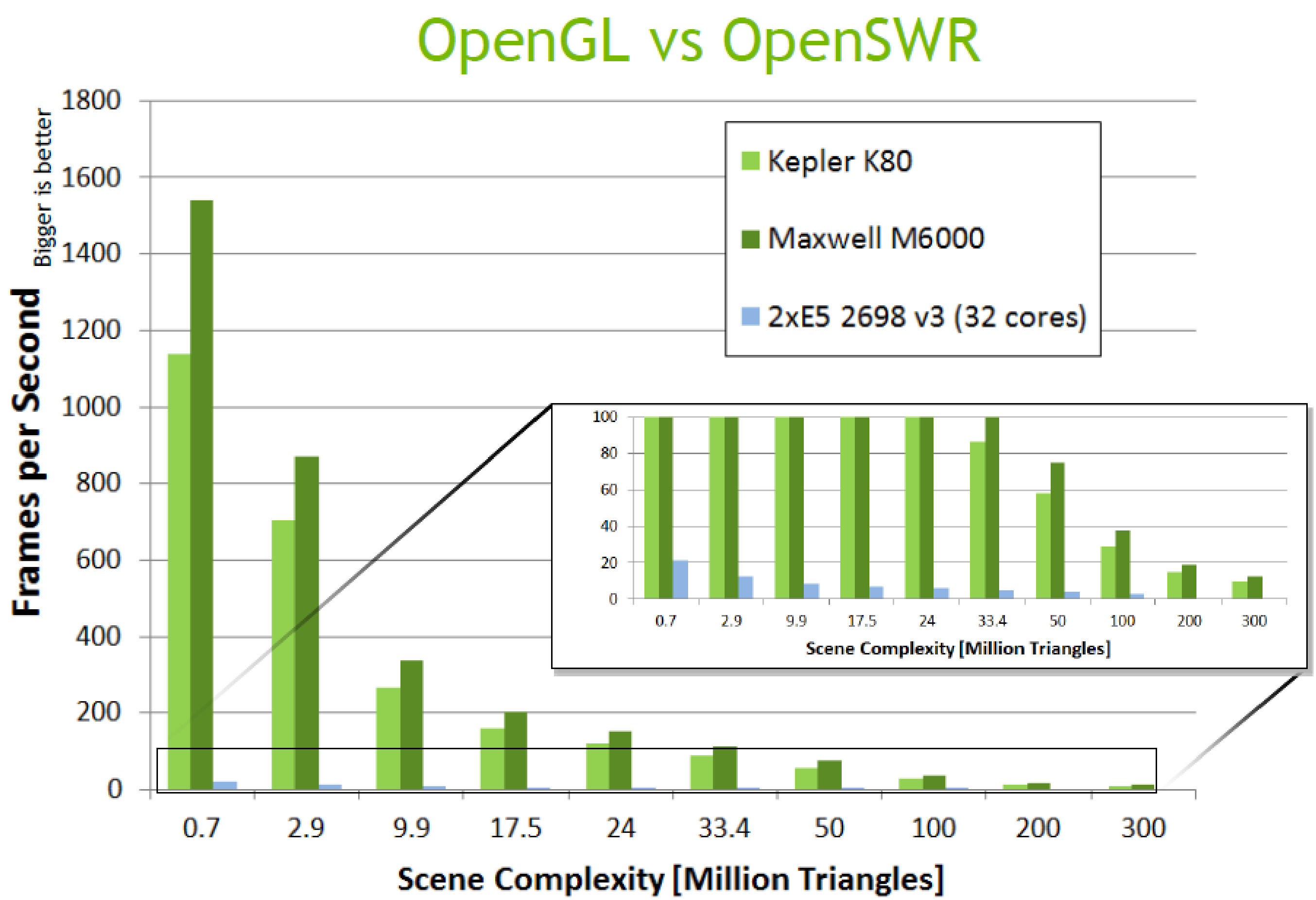


Rasterization Libraries

OpenSWR vs MESA LLVMpipe Performance Comparison
(drop in replacement for OpenGL)



Rasterization Libraries



*Scientific Visualization in HPC presentation from Peter Messmer (NVIDIA)

In-Situ Visualization

- But they agree that:

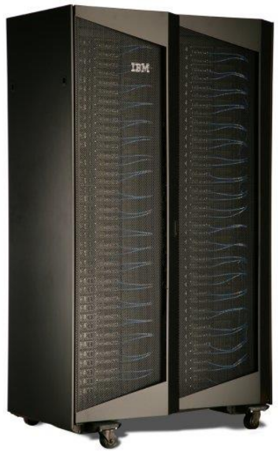
"Ideally you want to render in the same compute nodes"

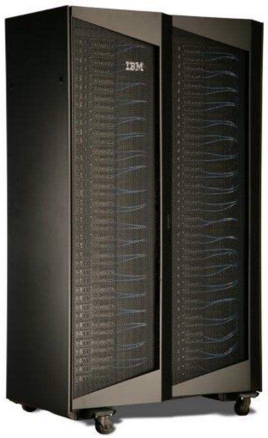
Compute + Visualization on the CPU

Compute + Visualization on the GPU

- In lots of workflows data movement is the most time (and energy) consuming step.

Hartree Centre main machine





Hartree Phase 1 (2012)

Blue Wonder – iDataPlex

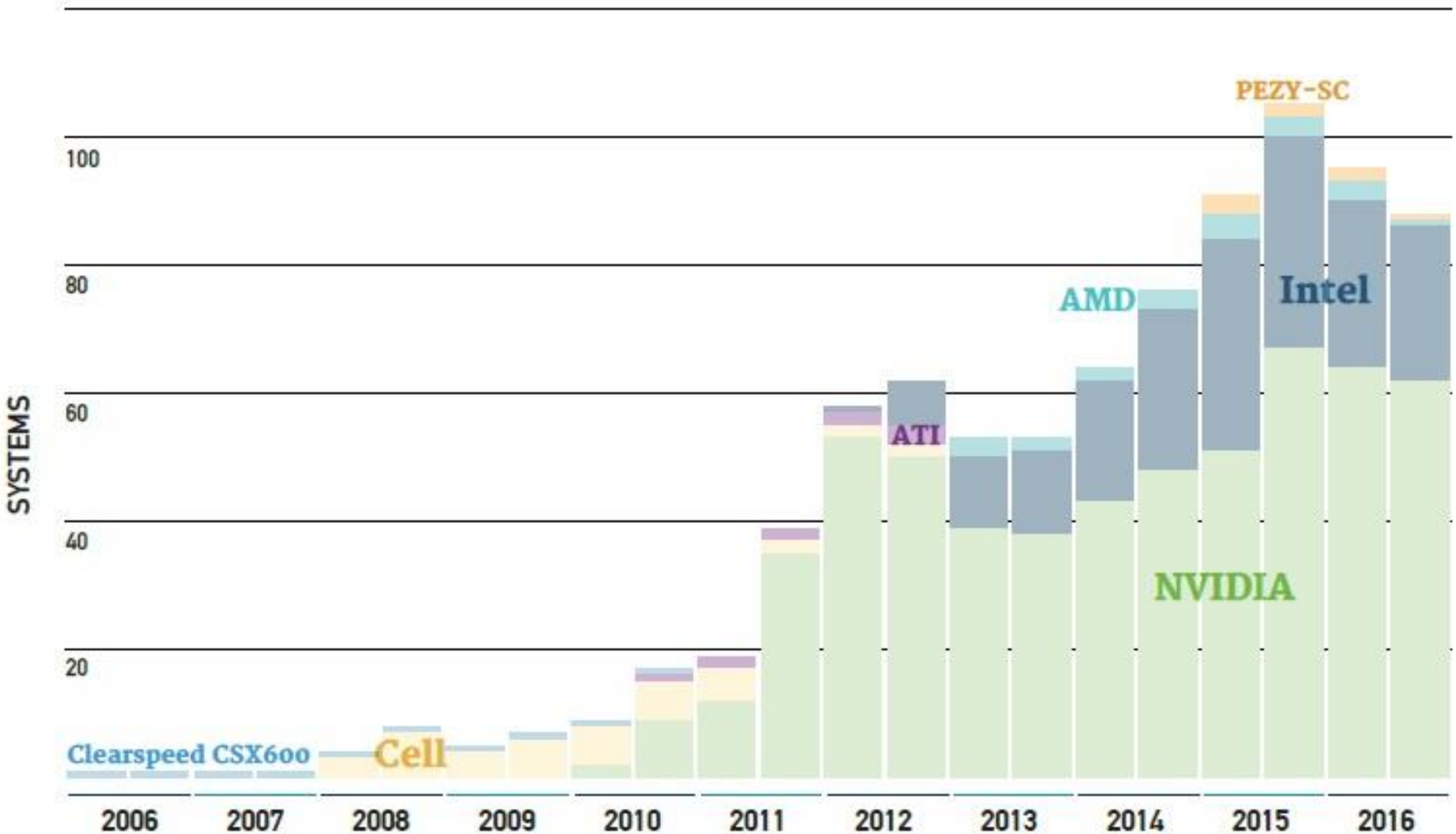
- 512 nodes
- 2 x 8 core 2.6 GHz
- Intel Xeon SandyBridge
- 32GB – 128GB RAM.
- Infiniband Interconnect
- **2 dedicated graphic nodes with nVidia Quadro 5000**

Hartree Phase 2 (2014)

Blue Wonder – iDataPlex

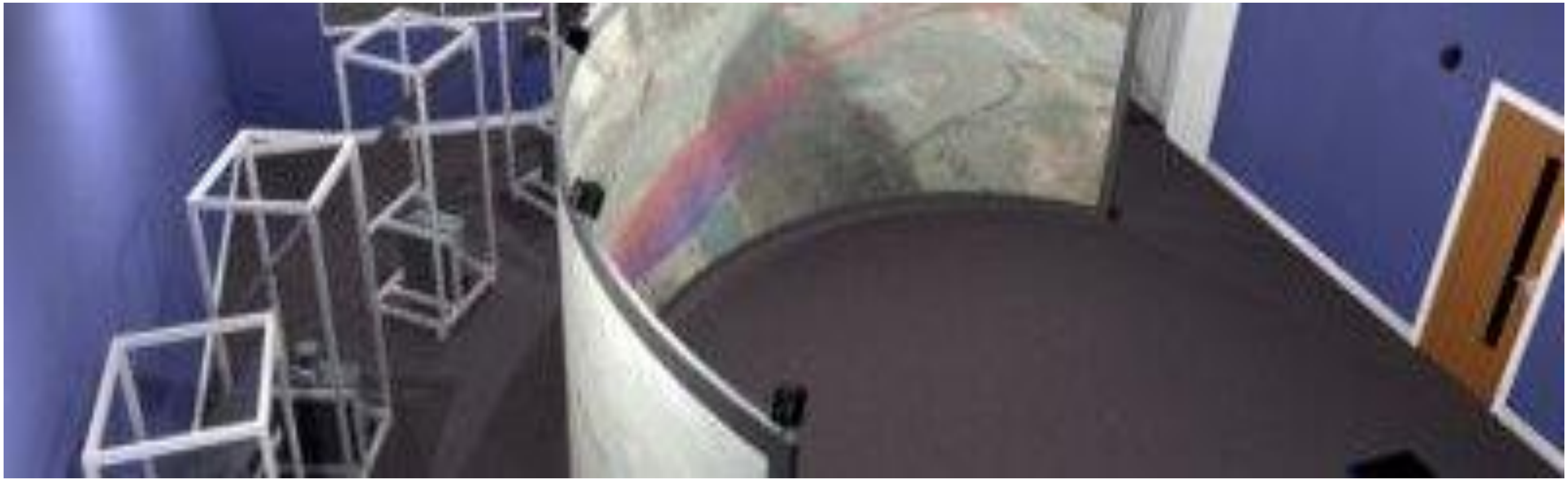
- 84 nodes
- 2 x 12 core 2.7GHz
- Intel Xeon IvyBridge
- 64GB RAM.
- Infiniband Interconnect
- **42 accelerators Intel Phi 5110P.**

Top 500


SYSTEMS
100
80
60
40
20
PEZY-SC
AMD
Intel
ATI
NVIDIA
Clearspeed CSX600
Cell
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016

Visualization Facilities

- Leverhulme 8025x1920, 10.25m by 2.3m; 15 megapixels a stereoscopic curved wall; blended to create one large interactive and collaborative space for data presentation and exploration.
- Crosfield (3615x1880, 5.5m by 2.85m; 6.6 megapixels) a stereoscopic rear-projected flat wall blended and integrated into an ideal project boardroom and meeting environment.



Section 2

RAY TRACING WITH OSPRAY

OpenGL

- Most of visualization nowadays is done with OpenGL, but it may not be the best choice for Scientific Visualization (any more):
  - why would I want to tessellate spheres/cylinders/etc to gazillions of triangles!?
  - why should I have to extract/store/render 100's of M's of tris to render an iso-surface?
  - do I really want to render 100's of M's of tris per frame? on a 1M pixel screen?
  - why is adding a bit of transparency such a big deal? Or volumes+surfaces!?
  - …

Rasterization vs. Ray Tracing

- Rasterization is fast. With specialized hardware for rasterization (GPUs) we can draw millions of polygons per second. But it works with a single triangle at a time.

- Ray tracers are doing more calculations that can access multiple parts of the model to achieve better (mathematical correctness) global illumination, light bouncing, shadowing, transparency, refractions and reflections.

Rasterization vs. Ray Tracing

Rasterized 3D graphics imagery

Ray traced 3D graphics imagery ©Siliconarts

OSPRAY Introduction

OSPRay is an **O**pen source, **S**calable, and **P**ortable **Ray** tracing engine for Intel Architecture CPUs.

- Ray tracing render in the processor.
- Developed by Intel
- Builds on top of Embree and ISPC
- Fully utilizes modern instruction sets like Intel SSE, AVX, AVX2, AVX512 or Intel® Xeon Phi
- http://ospray.github.io/

OSPRay Design Goals

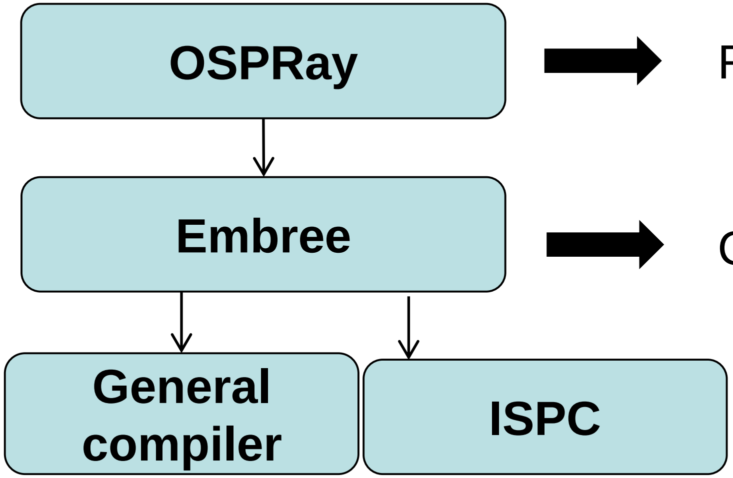
- Offer Compelling CPU rendering solution for Visualization
  - Target upcoming systems such as Stampede 2, Trinity, Cori, Theta, …
- Large data, volume rendering, … .
- Easy adoptability by actual end users
  - Free, have to integrate into commodity vis tools (ParaView, VisIt, VMD, …)
- Easy adoptability for tool developers
  - Must be easy to build, easy to integrate (many platforms, compilers, CPU types, …)
  - Easy to code to, understand, extend, …
  - API, open source

OSPRAY Introduction

Hartree Centre
Science & Technology Facilities Council

OSPRAY Introduction

Software stack

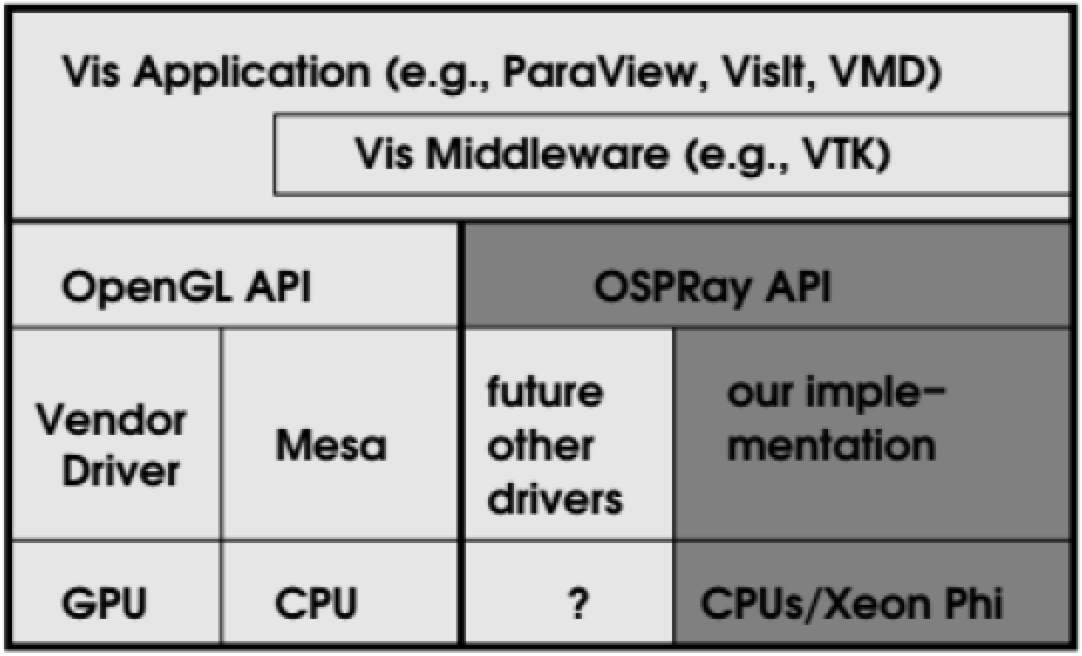


OSPRay → Ray tracing library for HPV on Intel CPUs

Embree → Collection of high-performance ray tracing kernels.

ISPC → C programming language with extensions for SPMD (hides the complexity of Vectorization & Multithreading)

OSPRay API



- http://www.sdvis.org/ospray/download/OSPRay_readme.pdf
- http://www.ospray.org/documentation.html

OSPRay API

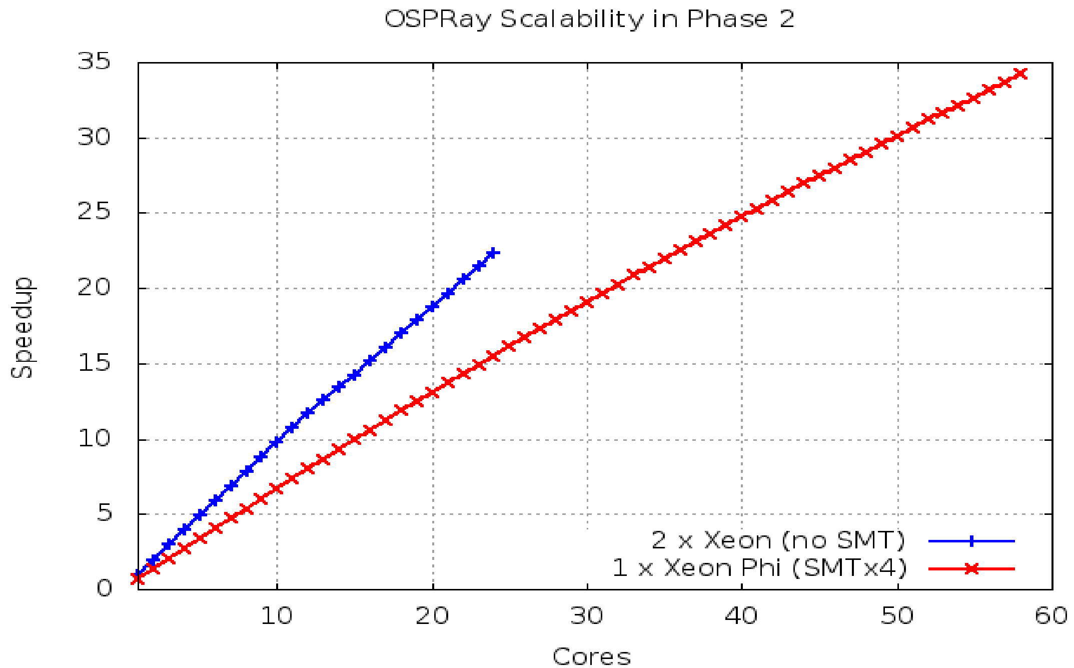
- Create actors
  - OSPGeometry spheres = ospNewGeometry("spheres");
- Create Data arrays (equivalent of GPGPU "buffers"; often zero-copy)
  - OSPDatacenter = ospNewData(N,OSP_VEC3F,&sphere Array[0]);
- Set parameters
  - ospSetData(spheres,"center",center);
- "commit" an object (ie, "apply those parameters")
  - ospCommit(spheres);
- Render frame…
  - ospRenderFrame(fb, renderer, OSP_FB_COLOR);
- … and map frame buffer
  - void * fb = ospMapFrameBuffer(fb,OSP_FB_COLOR);
  - glDrawPixels(… .)

OSPRay Reference Applications

- OSPRay comes with some Reference Applications that shows how to use the API and are useful for benchmarking the performance:
  - ospBenchmark
  - ospGlutViewer
  - ospQTViewer
  - ospVolumeViewer

Performance

- 2 x Xeon IvyBridge Performance:
  - Renders a 1024 x 1024 image of a 8.4M triangles model with 24 threads at **34.662 fps**.
- Intel Xeon Phi KnC vs Intel Xeon IvyBridge CPU:



OSPRay Examples

- See Intel IEEEVis2016 presentation from page 29 to 38

Section 1

HANDS-ON 1: INSTALLING AND USING OSPRAY

OSPRay Reference Apps on Xeons

$ **qsub –X –I –q edu**

- Install FreeGlut (prerequisite for the Reference applications)
  - $ wget -qO- "http://prdownloads.sourceforge.net/freeglut/freeglut-3.0.0.tar.gz?download" | tar xz
  - $ cd freeglut-3.0.0
  - $ cmake -DCMAKE_INSTALL_PREFIX=$HOME/freeglut
  - $ make –j 8 && make install
  - $ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$HOME/freeglut/lib64/
  - $ export LIBRARY_PATH=$LIBRARY_PATH:$HOME/freeglut/lib64/
  - $ cd ..
- Download pre-built OSPRay (http://www.ospray.org/getting_ospray.html)
  - $ wget –qO- https://github.com/ospray/OSPRay/releases/download/v1.2.0/ospray-1.2.0.x86_64.linux.tar.gz | tar xz
- Run OSPRay interactively on the Xeon
  - $ ./ospary<version>/bin/<application> [parameters] [file]

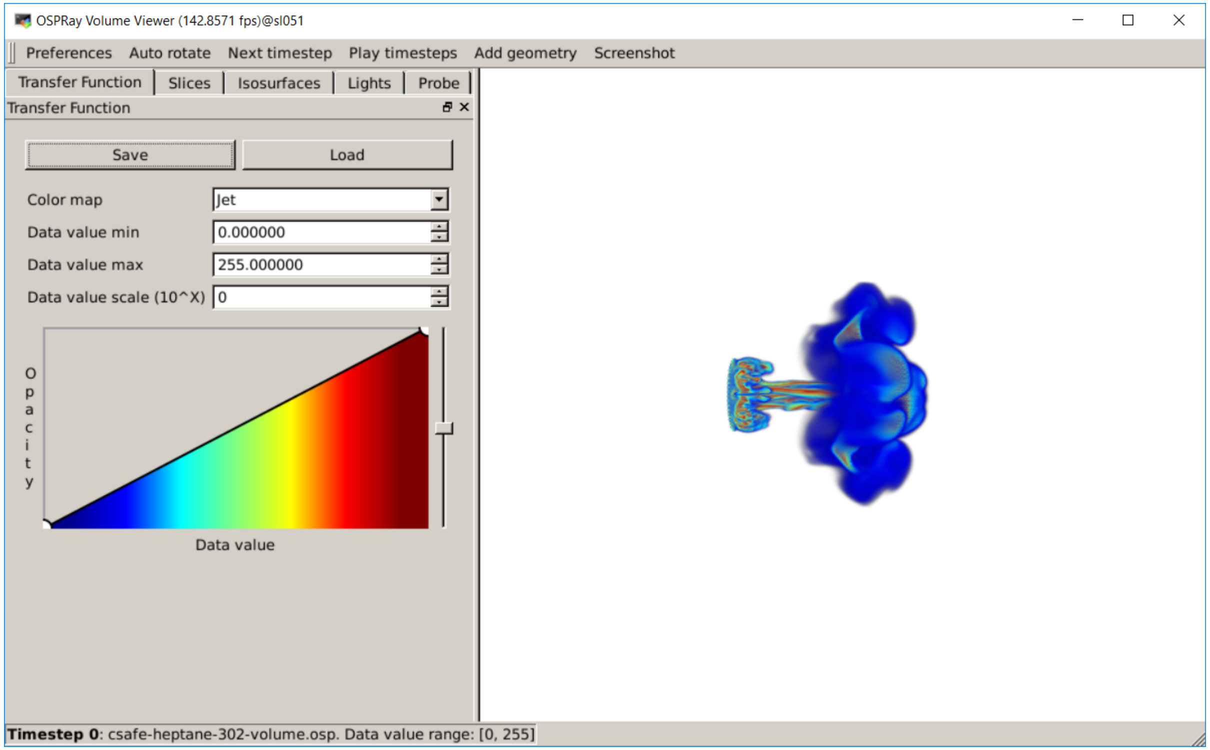
Hartree Centre
Science & Technology Facilities Council

OSPRay Reference Apps on KNC

- Install OSPRay for the Xeon Phi KNC
  - $ wget -qO- https://sourceforge.net/projects/ispcmirror/files/v1.8.2/ispc-v1.8.2-linux.tar.gz/download | tar xz
  - $ git clone https://github.com/ospray/ospray && cd ospray && **git checkout release-0.8**
  - $ source /opt/intel/parallel_studio_xe_2017.2.050/psxevars.sh
  - $ #comment line 192 on 'ospray/embree/common/sys/platform.cpp'
  - $ mkdir build && cd build
  - $ CC=icc CXX=icpc cmake -DOSPRAY_COMPILER=ICC -DOSPRAY_BUILD_MIC_SUPPORT=ON -DOSPRAY_BUILD_MPI_DEVICE=ON -DGLUT_glut_LIBRARY=<fg-installpath> -DGLUT_INCLUDE_DIR=<fg-install-path>/include ..
  - $ make –j 8
- Run OSPRay on the Xeon Phi KNL (Symmetric MPI)
  - $ source /opt/intel/parallel_studio_xe_2017.2.050/psxevars.sh
  - $ export I_MPI_MIC=1
  - $ mpirun -n 1 -host \`hostname\` ./ospVolumeViewer /home/sergisiso/data/Volumes/csafe-heptane-302-volume/csafe-heptane-302-volume.osp -showframerate : -n 1 -host \`hostname\`-mic0 ./ospray_mpi_worker.mic

Try the Reference Apps

- What is the performance of OSRPAY on the Xeon and Xeon Phi?
- You should be able to visualize the datasets under $HOME/../sergisiso/data/{Models|Volumes}



Section 2

REMOTE VISUALIZATION WITH PARAVIEW

Hartree Centre
Science & Technology Facilities Council

OSPRay is integrated into existing Vis tools

- ParaView: Ships since 5.1 (surfaces only), latest is ParaView5.2 (both)
- VisIt: prototypical OSPRay integration
- VMD: Latest version supports OSPRayrenderer
- VTK: Early integration by Dave DeMarle
- More Integrations now in early stages

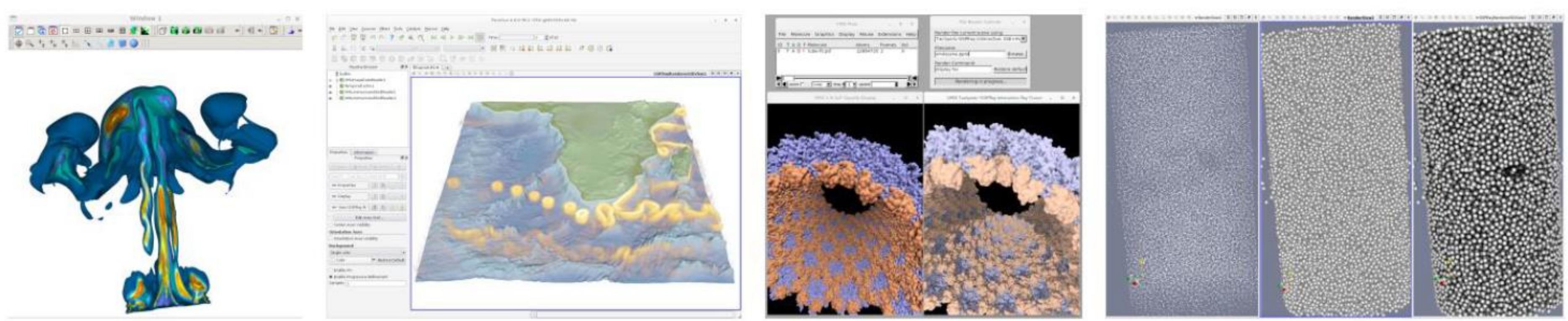
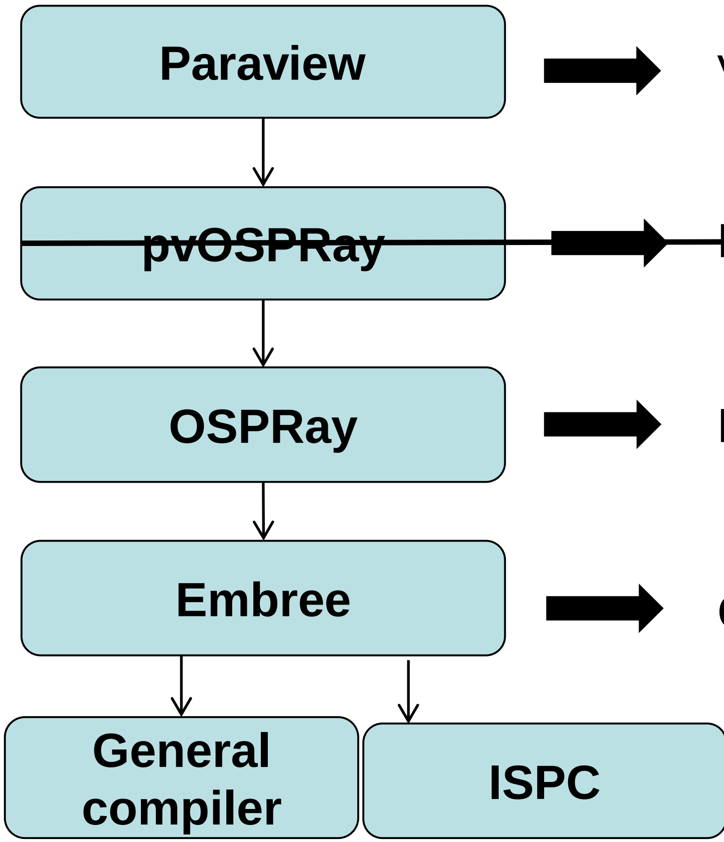


Fig. 8. Though still in Beta release, our OSPRay implementation is already prototypically integrated into three of the most widely used visualization tools, from left to right: VisIt; ParaView; VMD; a prototypical integration into VTK (done by Dave DeMarle at Kitware), showing a simple VTK application using three different VTK renderers—OpenGL points, GL Point Sprites, and OSPRay—side-by-side. Note the improvement in partical locality with ambient occlusion.

Paraview Integration

Software stack



Paraview → Visualization software

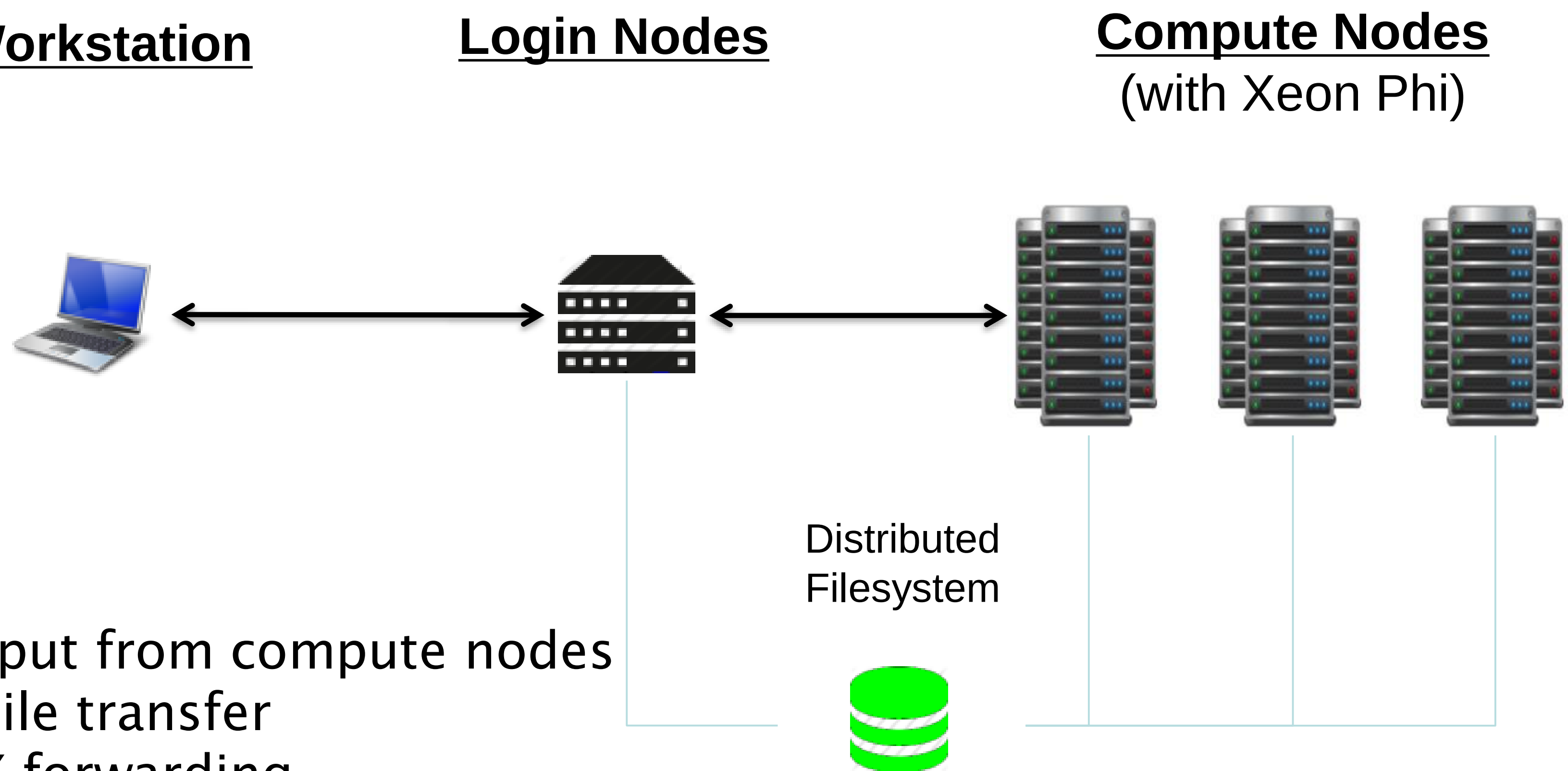
~~pvOSPRay → Paraview plugin to render with OSPRay~~

OSPRay → Ray tracing library for HPV on Intel CPUs

Embree → Collection of high-performance ray tracing kernels.

ISPC → C programming language with extensions for SPMD (hides the complexity of Vectorization & Multithreading)

Architecture



Output from compute nodes

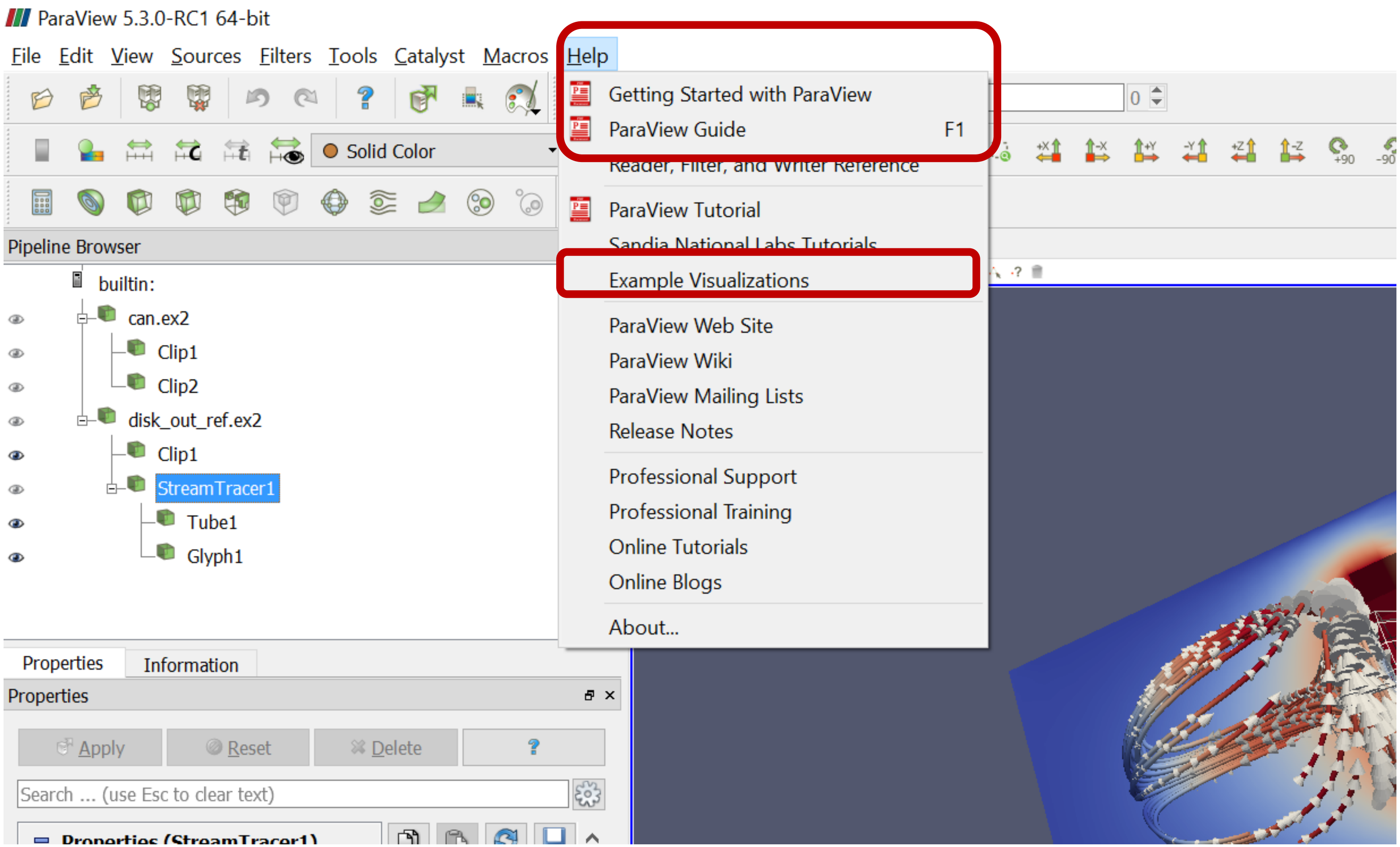
- File transfer
- X forwarding
- Remote rendering

Download and Install Paraview

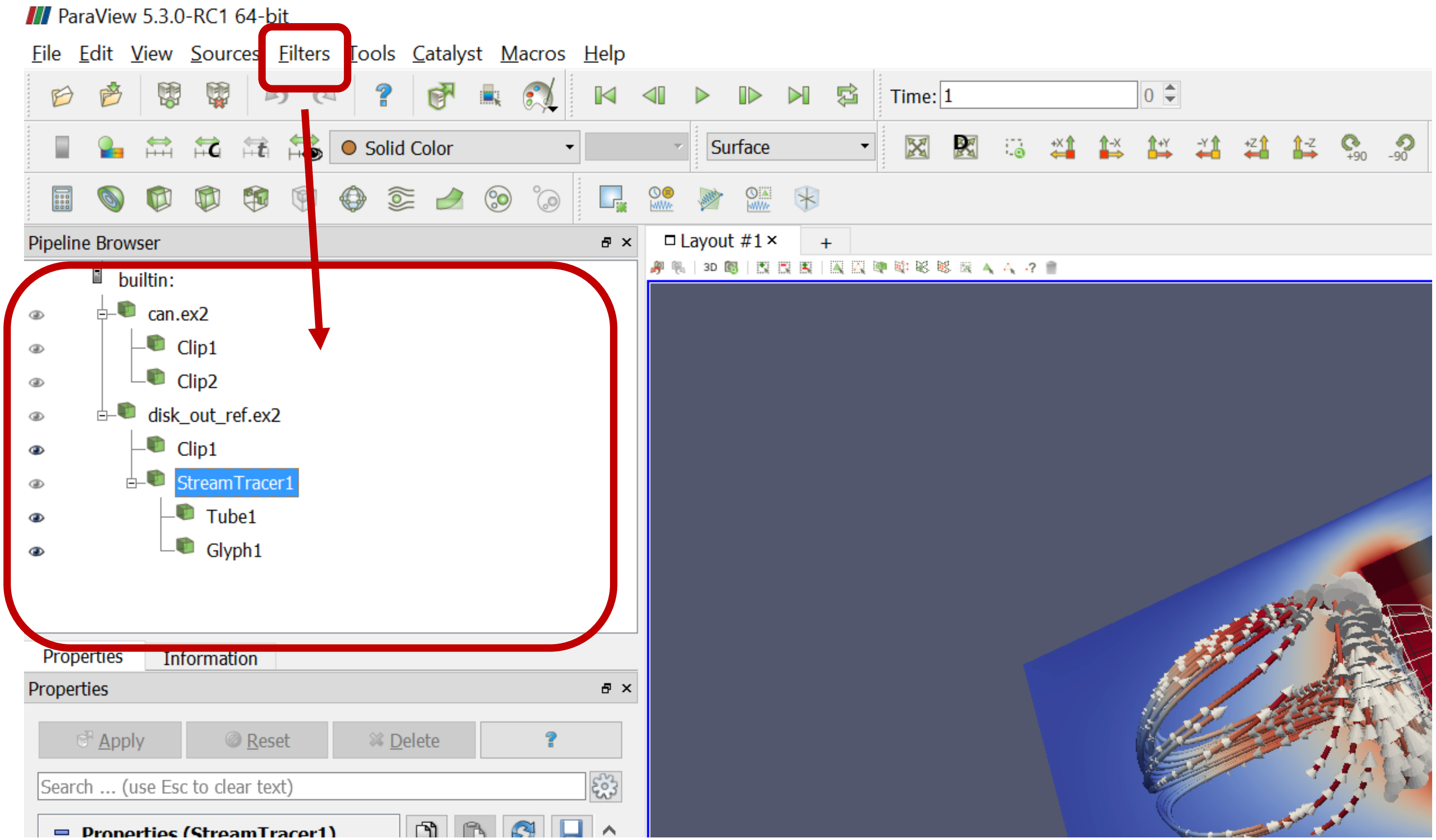
- Download Paraview on both local workstation and remote cluster
  - On your workstation go to http://www.paraview.org/download/ and choose the Linux/Windows/Mac installation

```
$ wget -qO- http://www.paraview.org/paraview-downloads/download.php?submit=Download&version=v5.3&type=binary&os=linux64&downloadFile=ParaView-5.3.0-RC1-Qt5-OpenGL2-MPI-Linux-64bit.tar.gz | tar xz
```

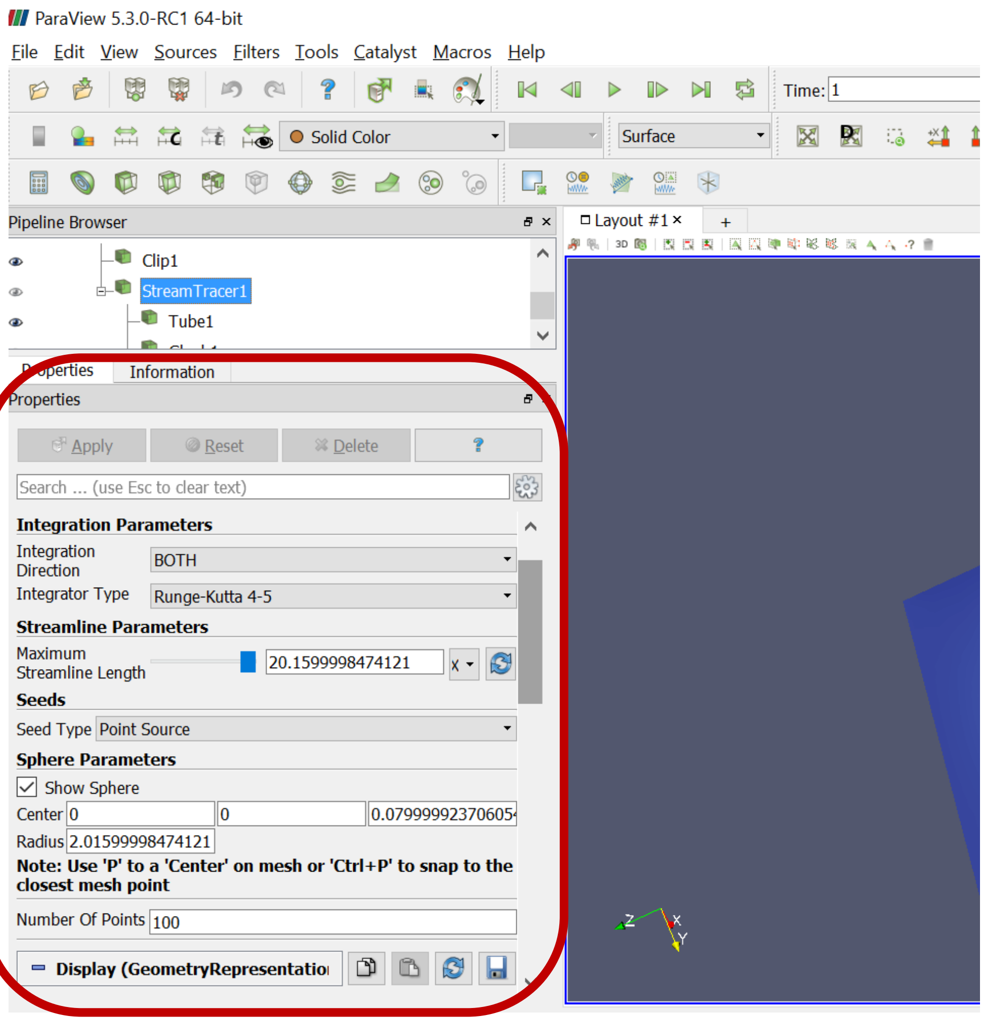
ParaView Introduction



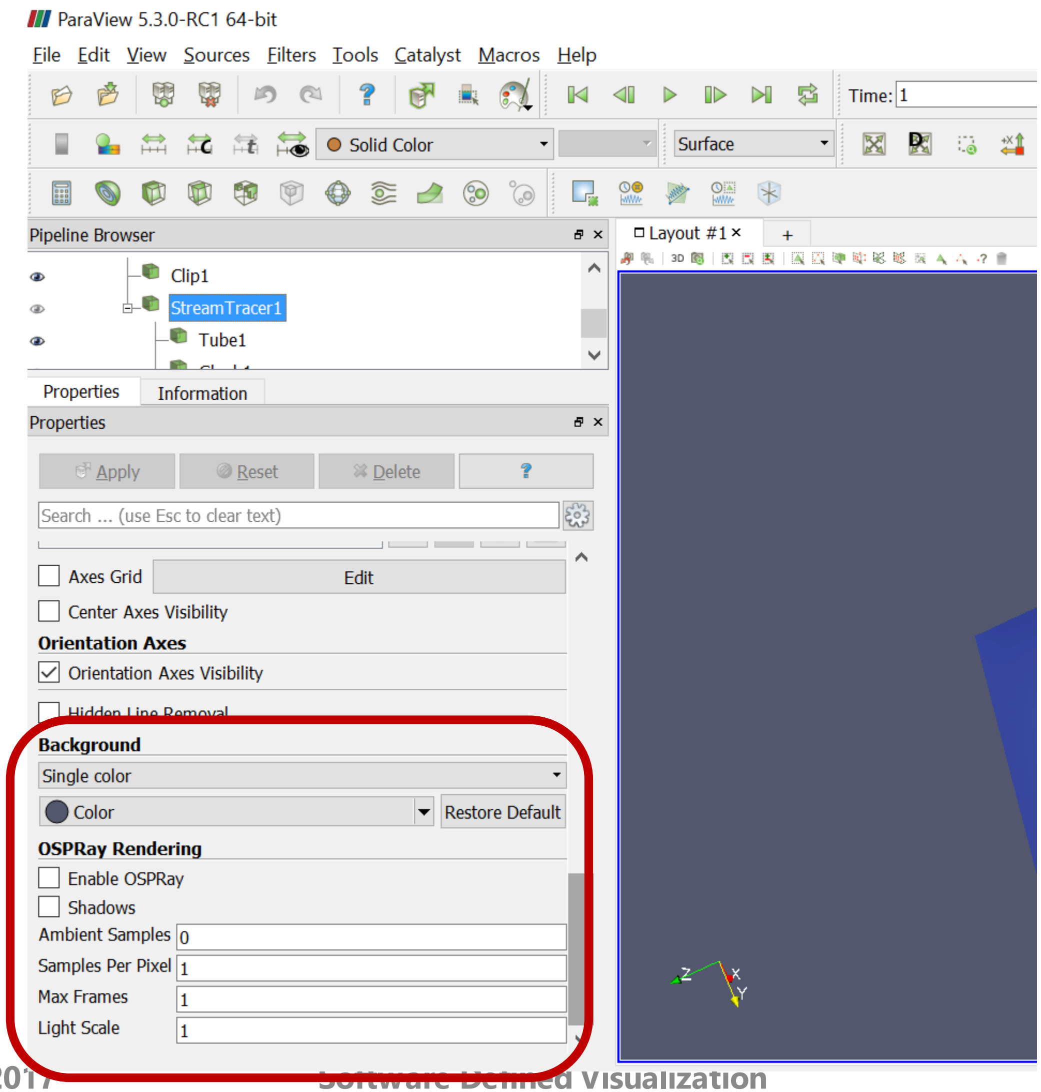
Pipeline of operations (filters)



Each filter has a set of properties



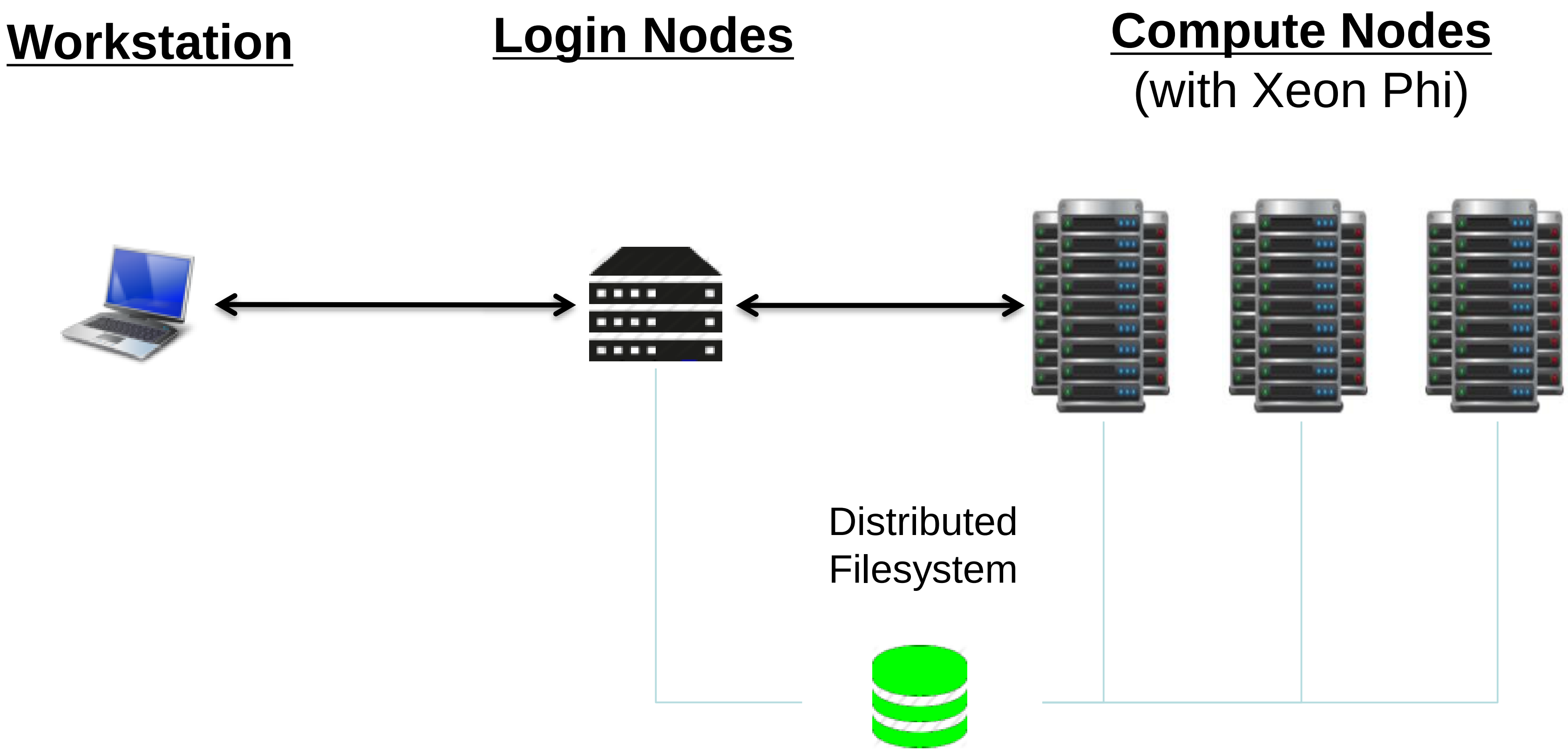
Scroll down to find Vis Properties (OSPRay)



Section 5

HANDS-ON 2: REMOTE VIS. OF OPENCH AND HVAC

Architecture



Launch server

File : start_server.sh

```
#!/bin/bash
#PBS -l nodes=1:ppn=4
#PBS -l walltime=00:30:00
source /opt/intel/parallel_studio_xe_2017.2.050/psxevars.sh
mpirun -n 4 $HOME/ParaView-5.3.0-RC1-Qt5-OpenGL2-MPI-Linux-64bit/bin/pvserver --
use-offscreen-rendering
```

Remote cluster

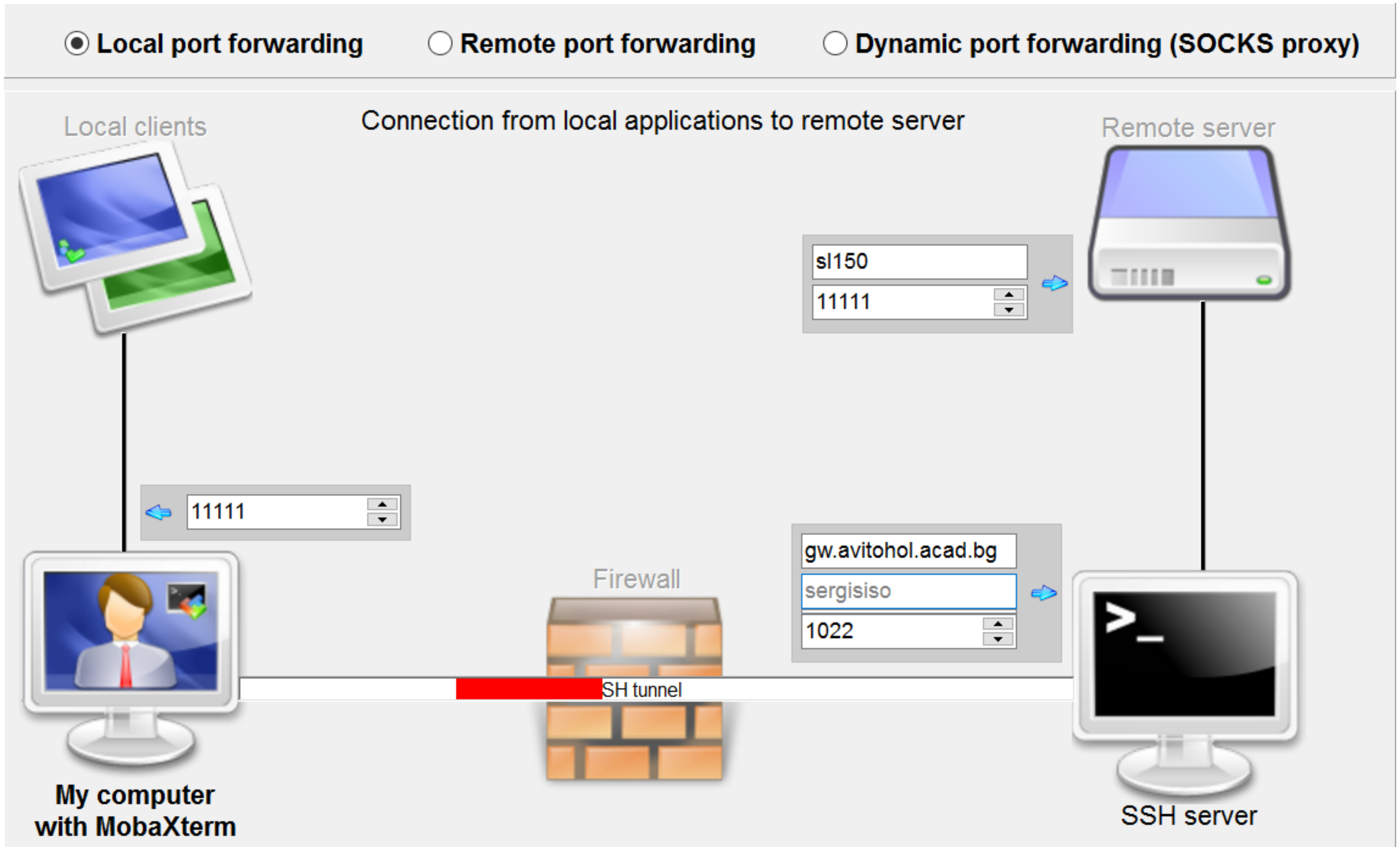
```
$ qsub start_server.sh
22478.moab
$ qstat -n 22478.moab
Req'd       Req'd       Elap
Job ID                  Username    Queue    Jobname          SessID  NDS   TSK
Memory      Time    S   Time
----------------------- ----------- -------- ---------------- ------ -----
22478.moab              sergisiso   batch    start_server.sh   57870     1
4       --    00:30:00 R  00:00:26
   sl150/0-3
```

Client System

```
$ ssh -N -L 11111:sl150:11111 <username>@gw.avitohol.acad.bg:1022
```

Hartree Centre
Science & Technology Facilities Council

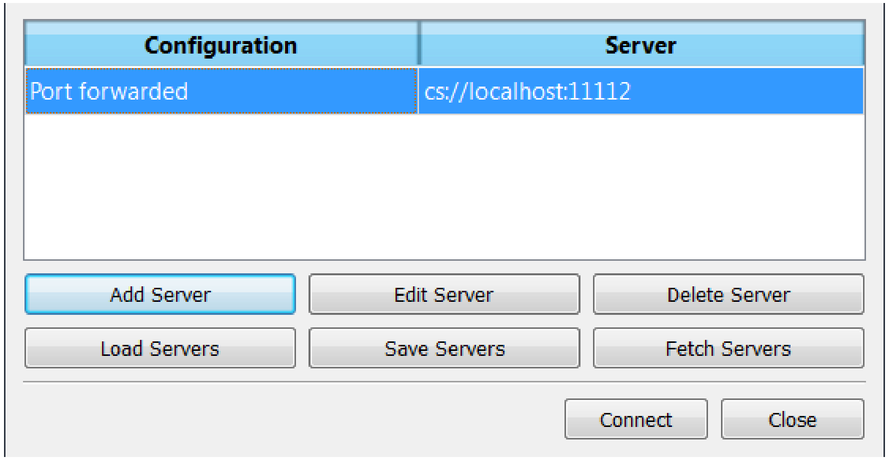
SSH Port Forwarding



$ ssh -N -L <local-port>:<compute-node>:<remote-port> <myid>@<machine-login>

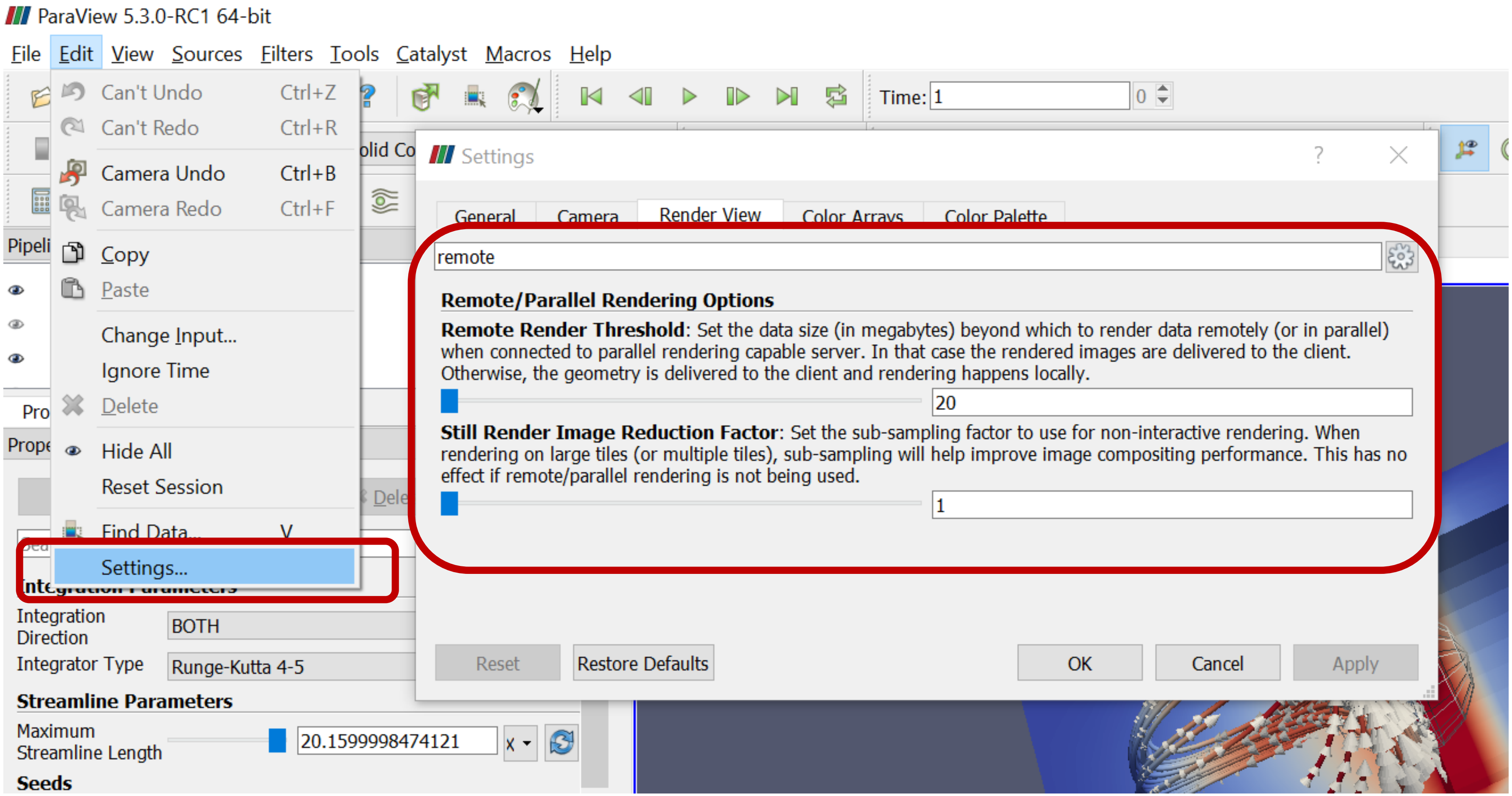
Connect Remote Paraview

- File > Connect ..



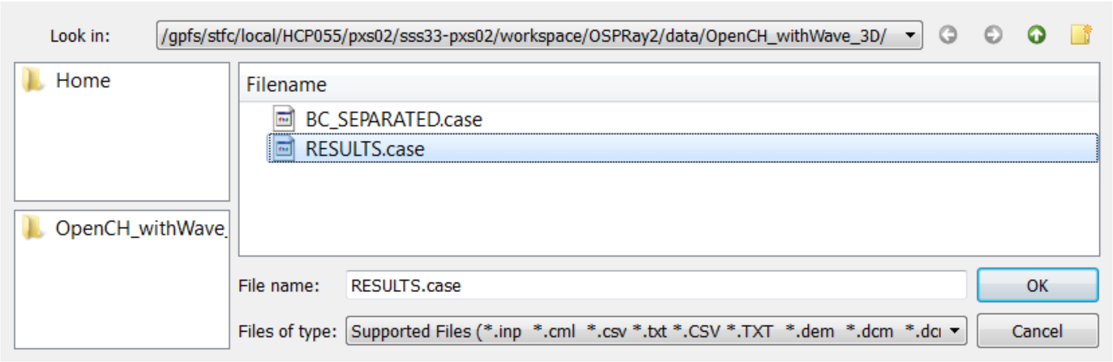
- Edit > Settings > Render View (for better performance ☺)
  - Set 'Remote Render Threshold' to 0 Mb (slider to the left)
  - Set 'Image Reduction Factor' to 4 or more (during interactions)
  - Optional: 'Image Compression' and 'Still Render Image Reduction Factor'

Remote Rendering Settings



Open remote files

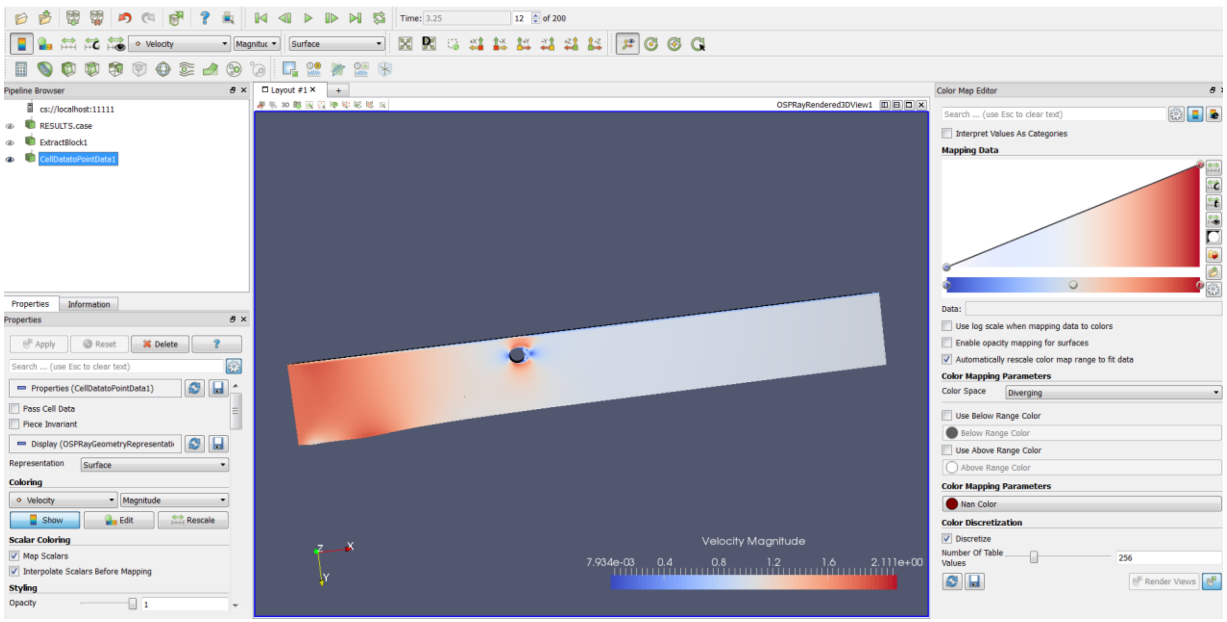
- Once connected File > Open menu option will show the remote file system.



Hartree Centre
Science & Technology Facilities Council

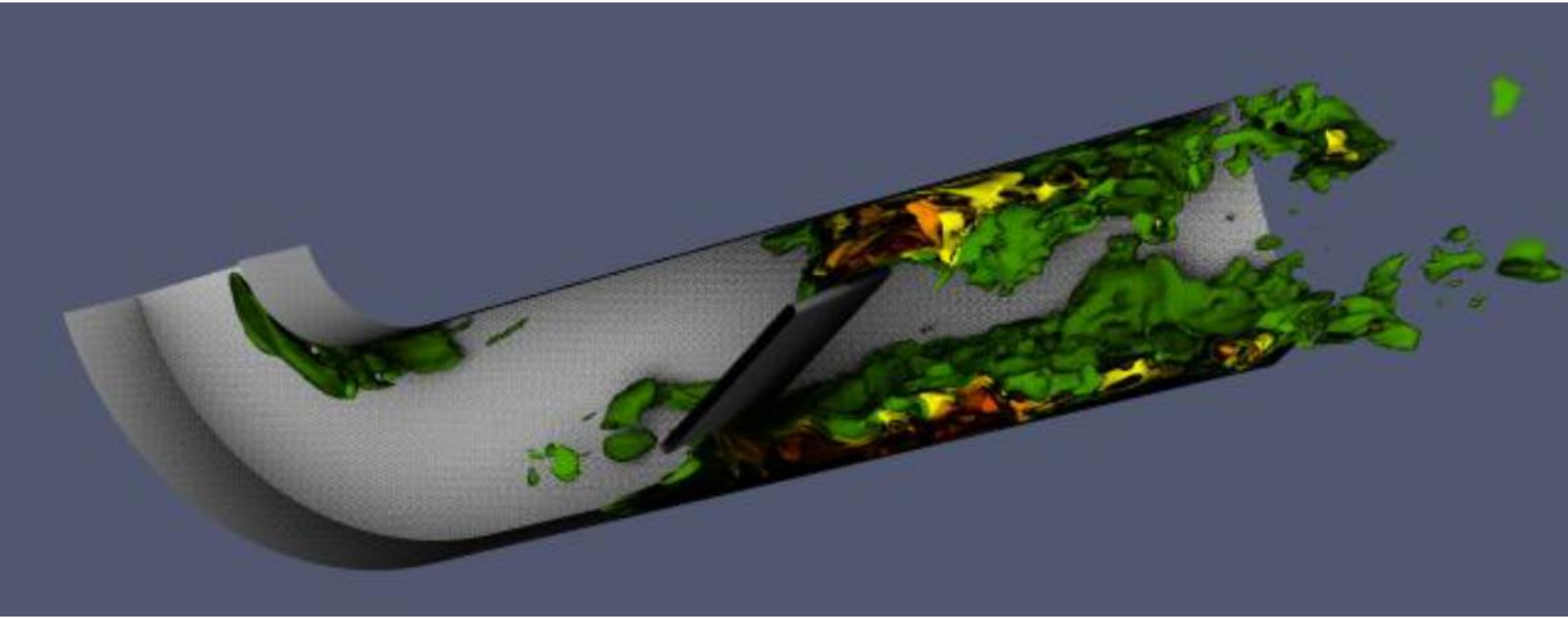
Load OpenCH 3D

- Apply Filters:
  - Filters > Extract Block
  - Filters > Cell Data to Point Data
- Toolbar > Play



HVAC Example

- Can you visualize HVAC isosurfaces? How much time does it take? How well it scales with number of server MPI processes?
  - Hint: Use ExtracBlock, CellToDataPoint, Array Calculator to get the vector module, Gradient and Countor Filters to generate the isosurfaces.

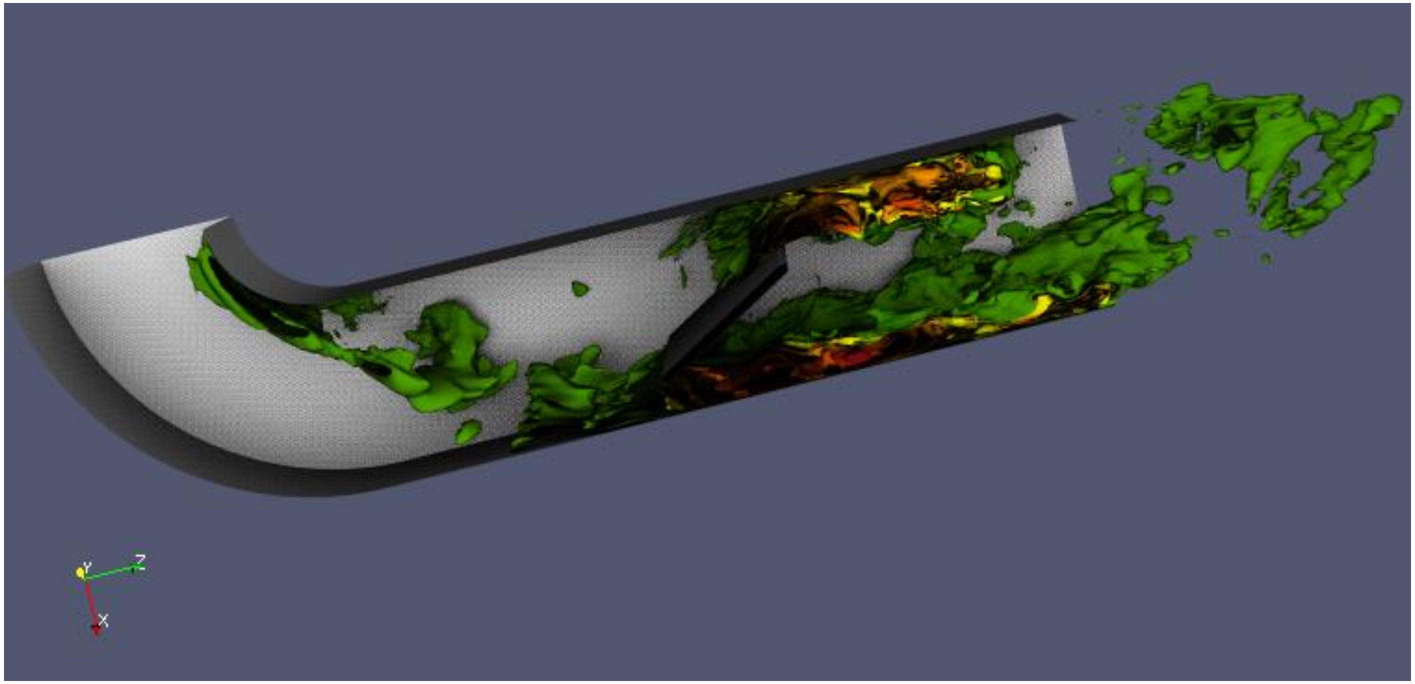


Section 3

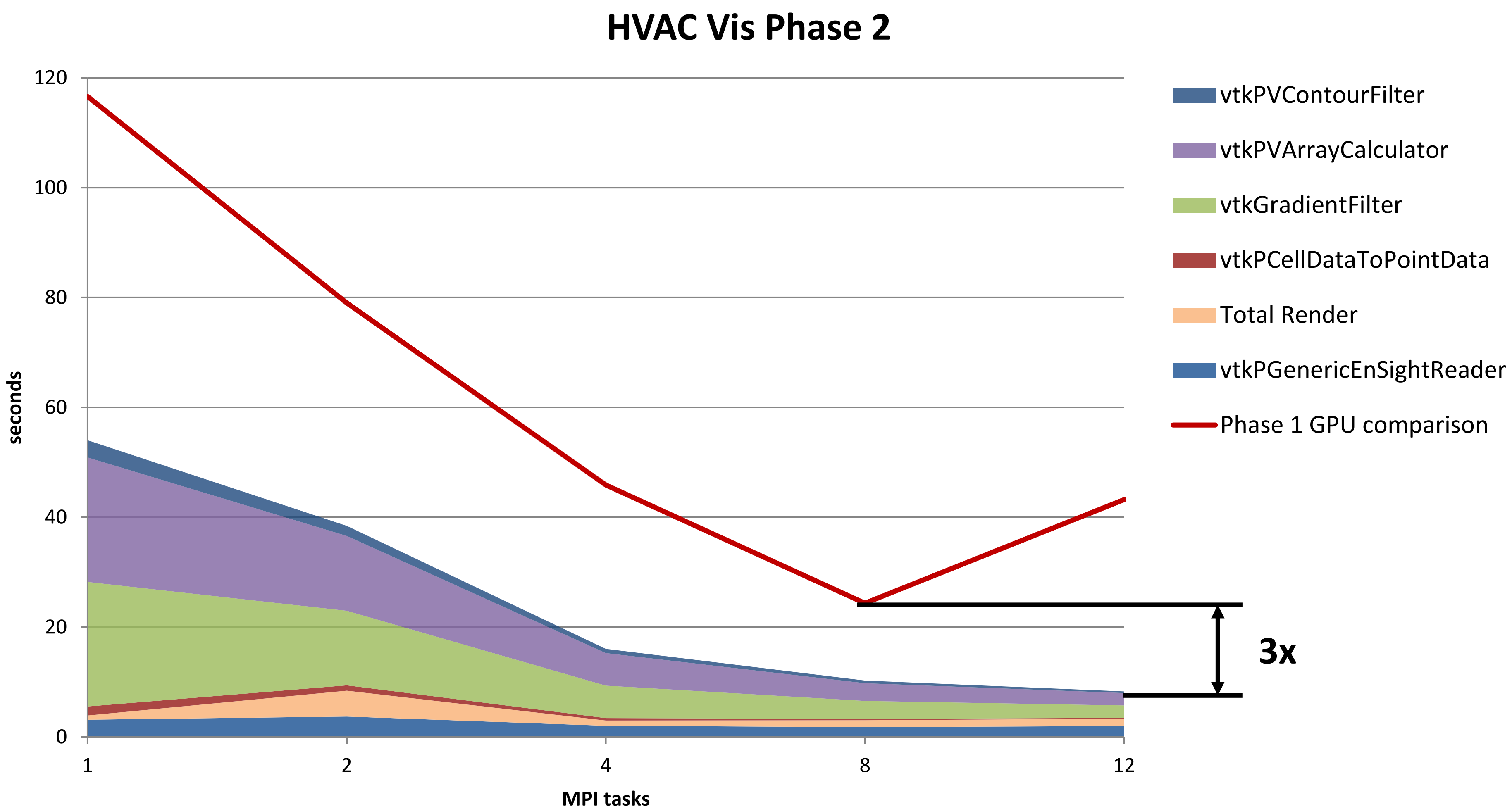
USE CASE 1: HVAC SIMULATION

HVAC Introduction

- CFD Simulation: Blend + Wall + Obstacle + Fluid
- Data: 150MB x 2000 time-steps -> 300Gb
- Visualization through Paraview: Interested in the Velocity Gradient

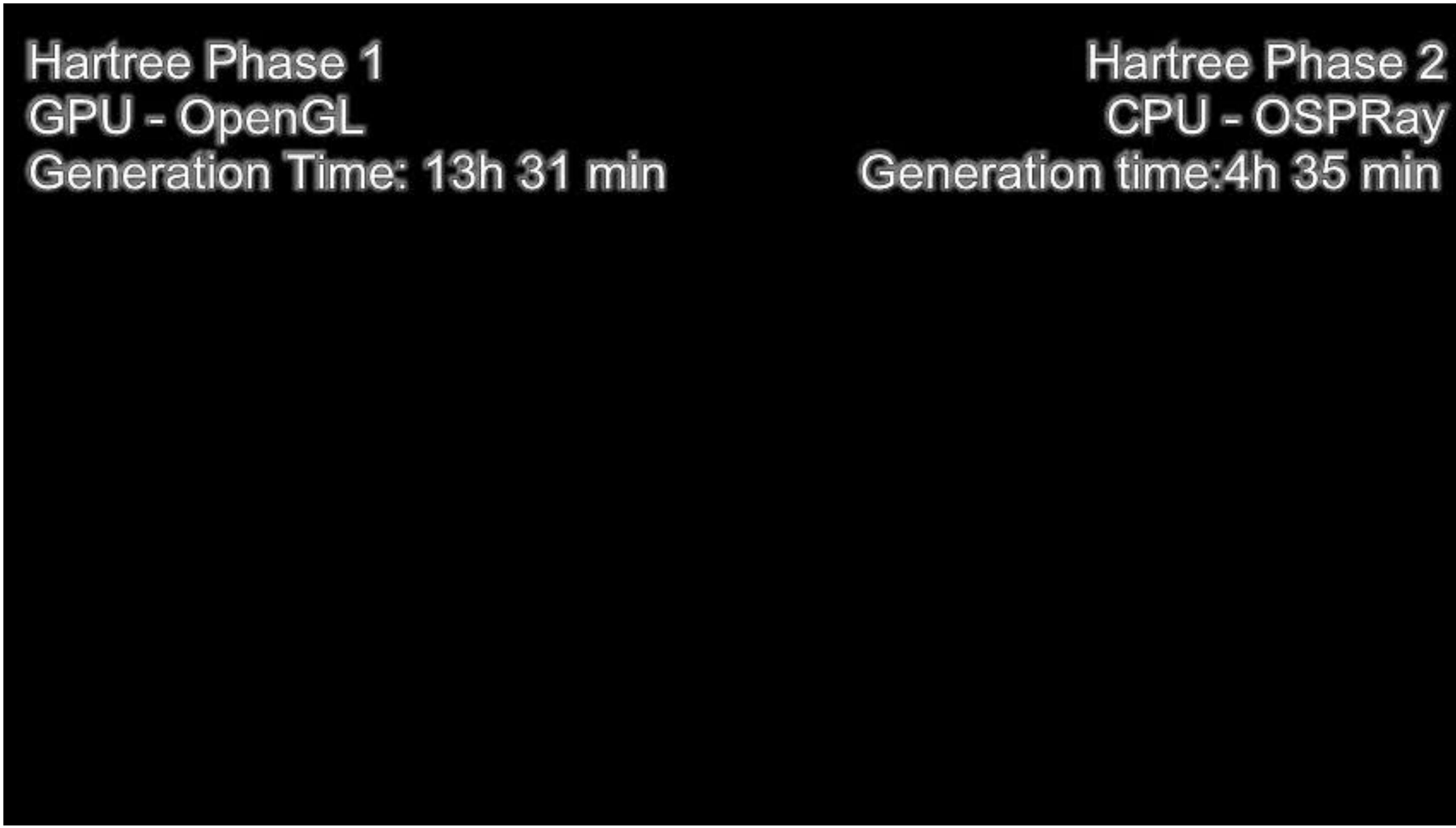


Performance

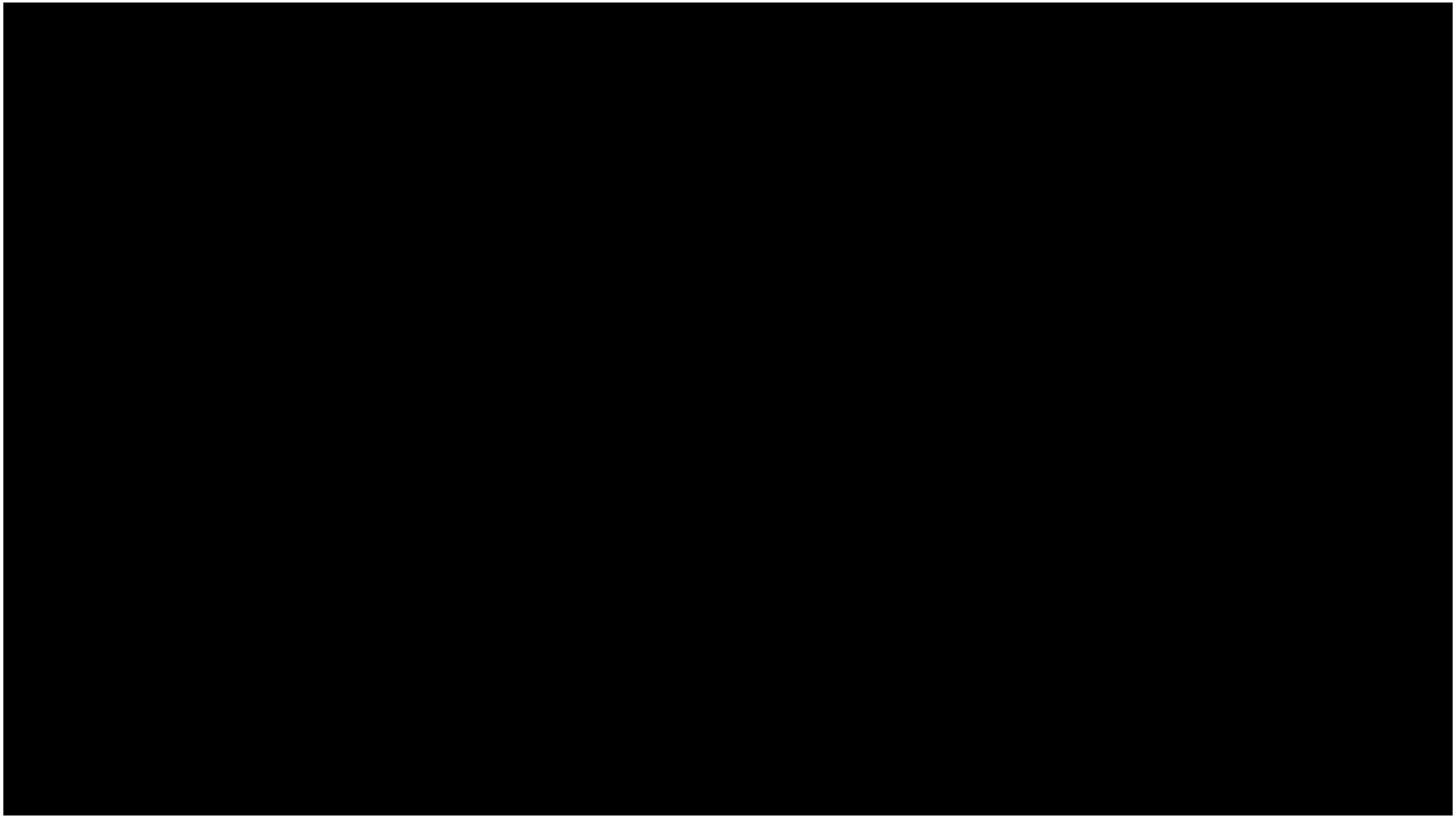


Hartree Centre
Science & Technology Facilities Council

HVAC Animation



HVAC Animation



Section 4

USE CASE 2: IMAT FACILITY

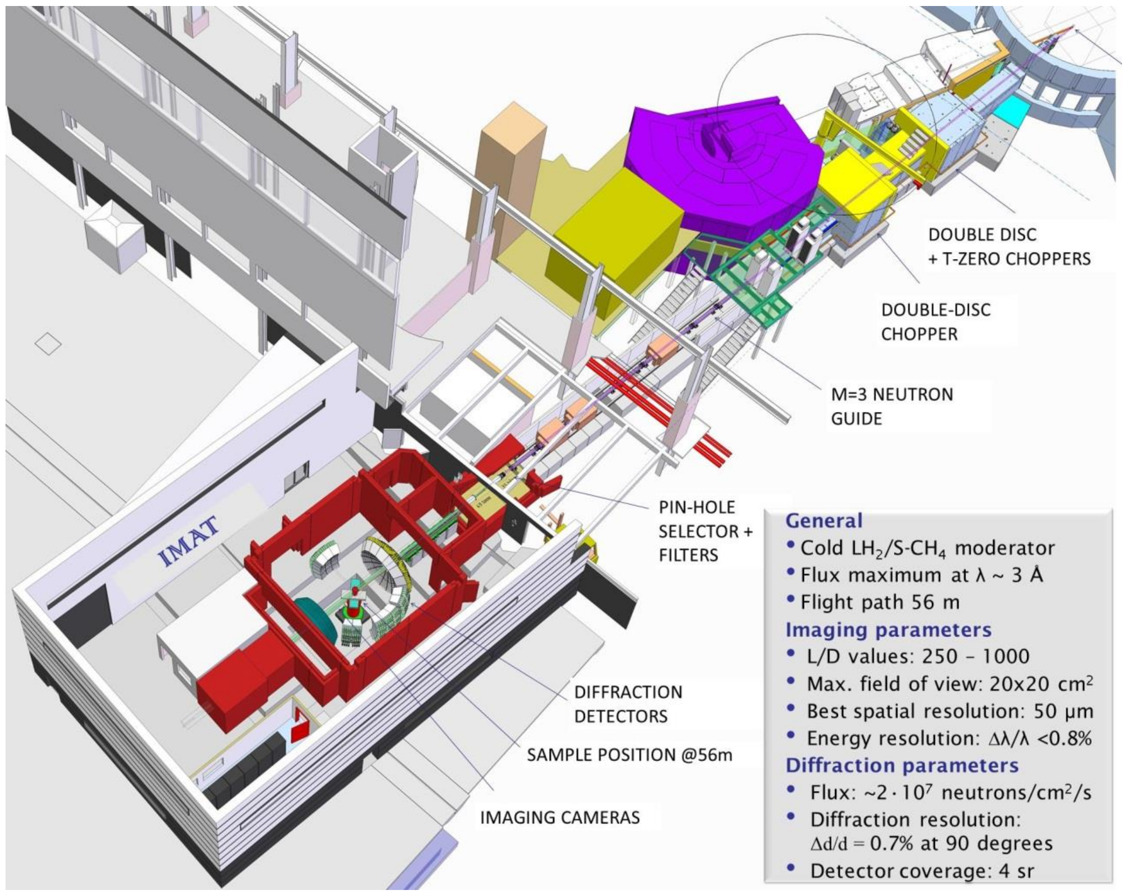
Data provided by Srikanth Nagella

Hartree Centre

Science & Technology Facilities Council

IMAT (ISIS) Facility

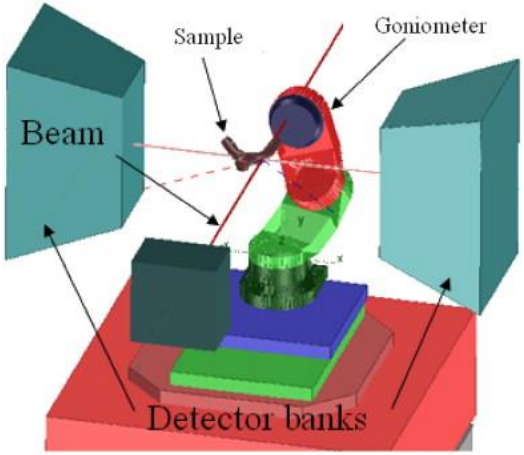
- A neutron imaging and diffraction instrument for materials science, materials processing and engineering.



Tomography-driven diffraction

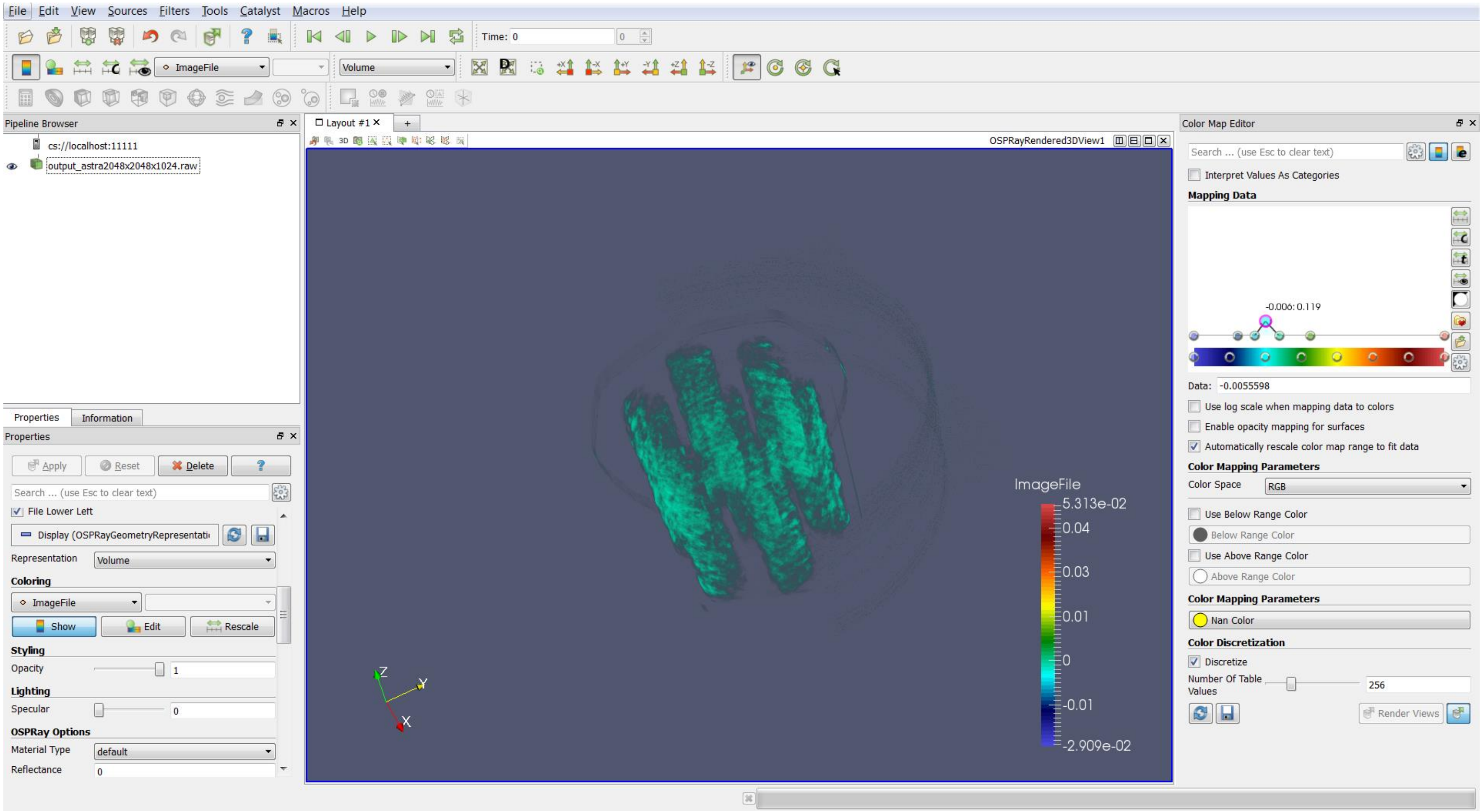
- Some properties can be more effectively analysed if the diffraction scans are guided by radiographic data.
- Generated data is up to 300 Gb
- Need results in less than 20 minutes

⇨ **in-situ visualization**



References:
[1] W. Kockelmann, G. Frei, E.H. Lehmann, P. Vontobel, J.R. Santisteban, *Energy-selective neutron transmission imaging at a pulsed source.*
[2] G. Burca, J.A. James, W. Kockelmann, M.E. Fitzpatrick et al., *A new bridge technique for neutron tomography and diffraction measurements.*

Current status



Section 7

QUESTIONS

Hartree Centre
Science & Technology Facilities Council