Large Scale Accelerated Rendering on 10K Ray Tracing Enabled Nodes

Roba Binyahib & David DeMarle

Graphics Software Engineer @ Intel

Supercomputing 2023

it starts with intel



Large Scale Accelerated Rendering on 10K Ray Tracing Enabled Nodes

intel

Large Scale Accelerated Rendering on 10K Ray Tracing Enabled Nodes



Rendering

Computing

Product Design



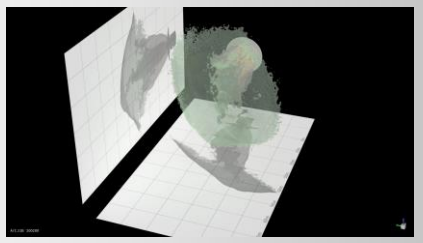
Courtesy of Bentley Motors

Film



The Addams Family 2 - Image courtesy Cinesite & MGM

Science



Courtesy of NASA Ames, Patrick Moran

Architecture



Intel Architecture Day 2021 Xᵉ Demo
4004 Moore Lane, Noyce Grove, CA

Gaming



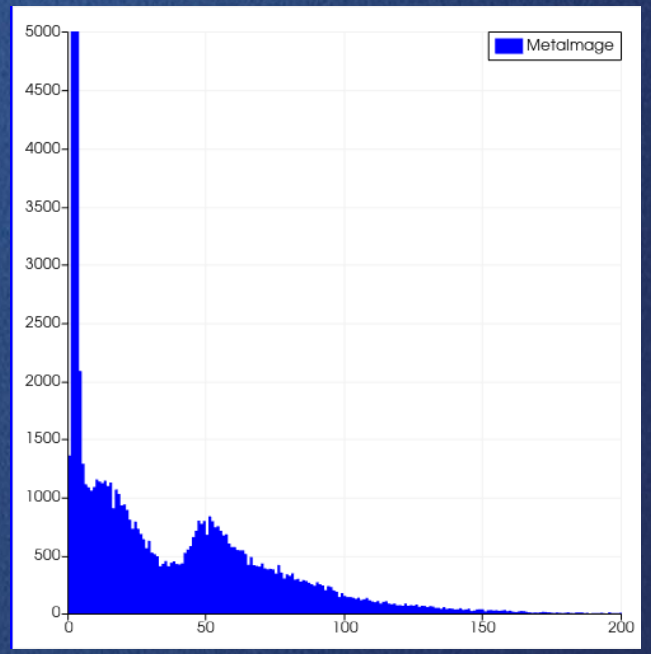
Copyright Activision Publishing

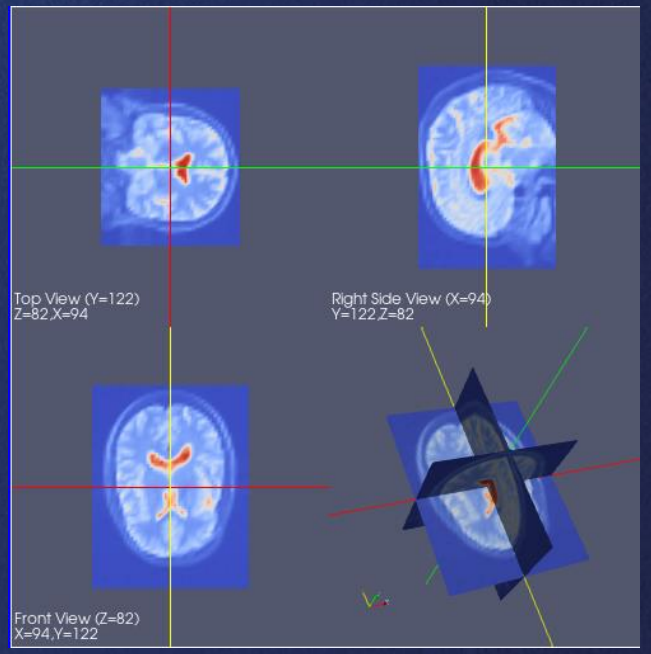
What is Scientific Visualization?

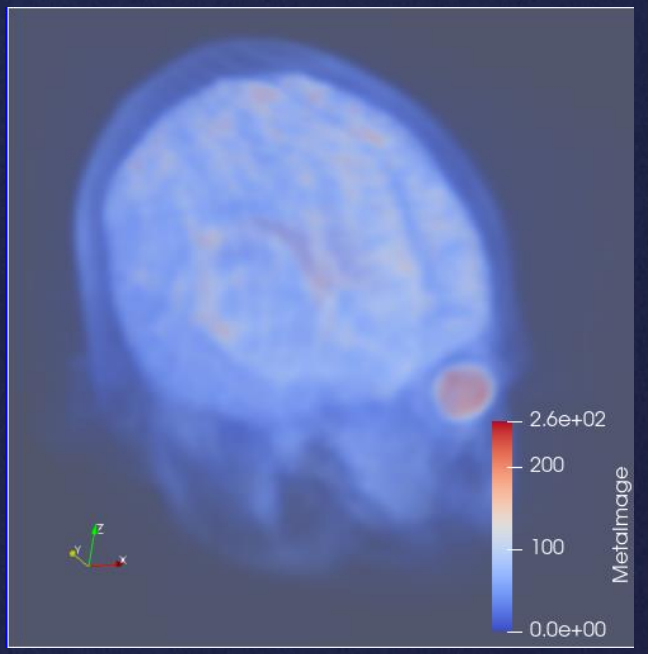
Numbers → plots, graphs, and pictures for 3D domains

Showing HeadMRVolume.mhd Attribute: Point Data

	Point ID	Structured Coordinates	MetaImage
72846	72846	30, 29, 24	67
72847	72847	31, 29, 24	93
72848	72848	32, 29, 24	59
72849	72849	33, 29, 24	47
72850	72850	34, 29, 24	46
72851	72851	35, 29, 24	48
72852	72852	36, 29, 24	60
72853	72853	37, 29, 24	138
72854	72854	38, 29, 24	105
72855	72855	39, 29, 24	15
72856	72856	40, 29, 24	17
72857	72857	41, 29, 24	41
72858	72858	42, 29, 24	31
72859	72859	43, 29, 24	4
72860	72860	44, 29, 24	2



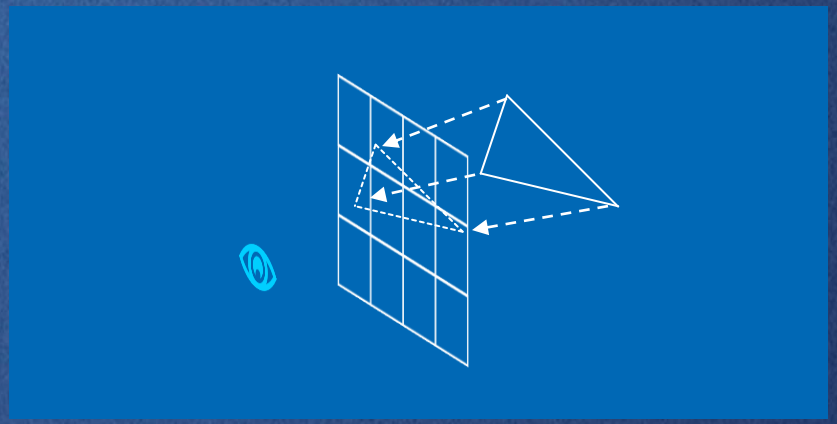


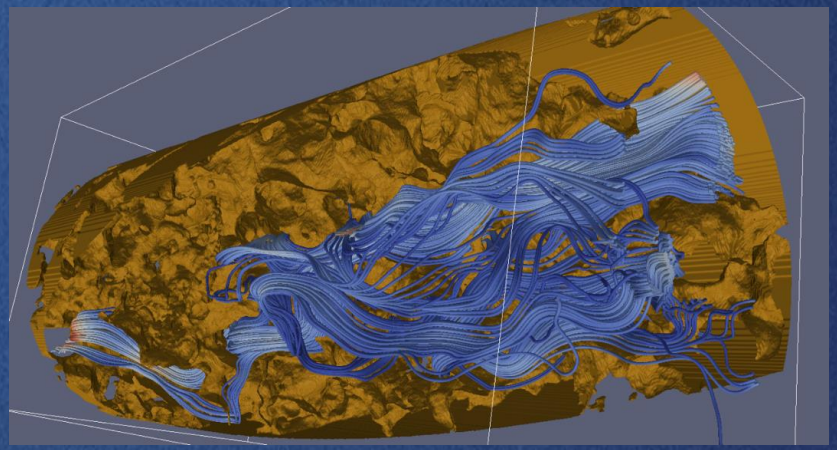


Rasterization

for each object

color covered pixels when object is nearest

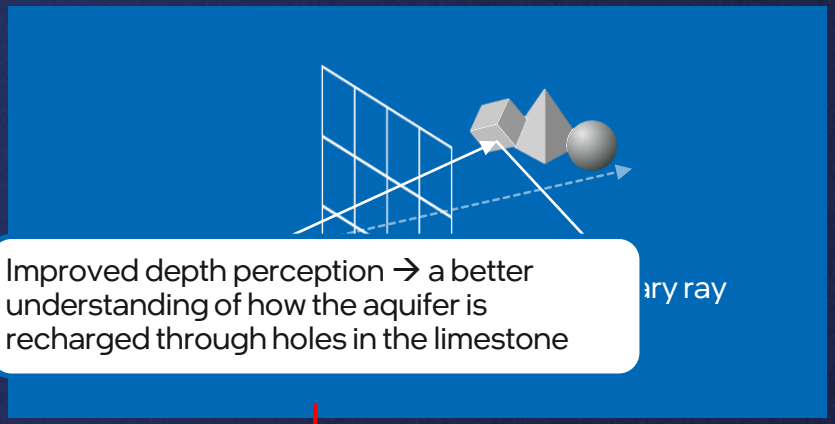


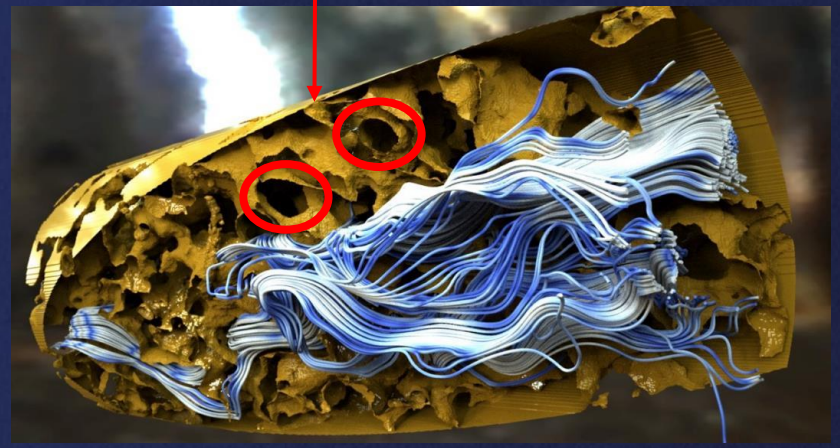


Ray Tracing

for each pixel

color from nearest object along ray



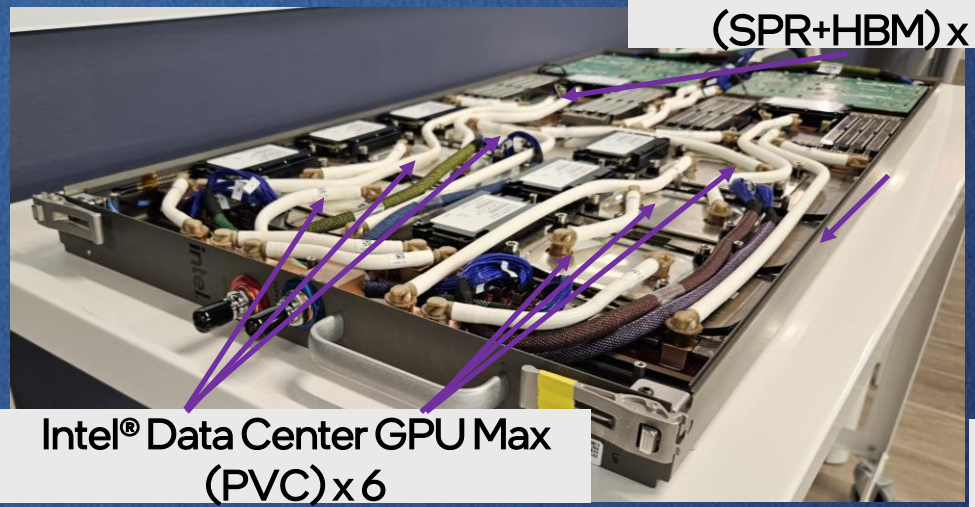


Visualizing water flowing through a limestone karst from a South Florida ground core sample. Credit: Data courtesy of Michael Sukop, Sade Garcia, Florida International University and Kevin Cunningham, United States Geological Survey. Visualization: Carson Brownlee, Aaron Knoll, Paul Navratil, Texas Advanced Computing

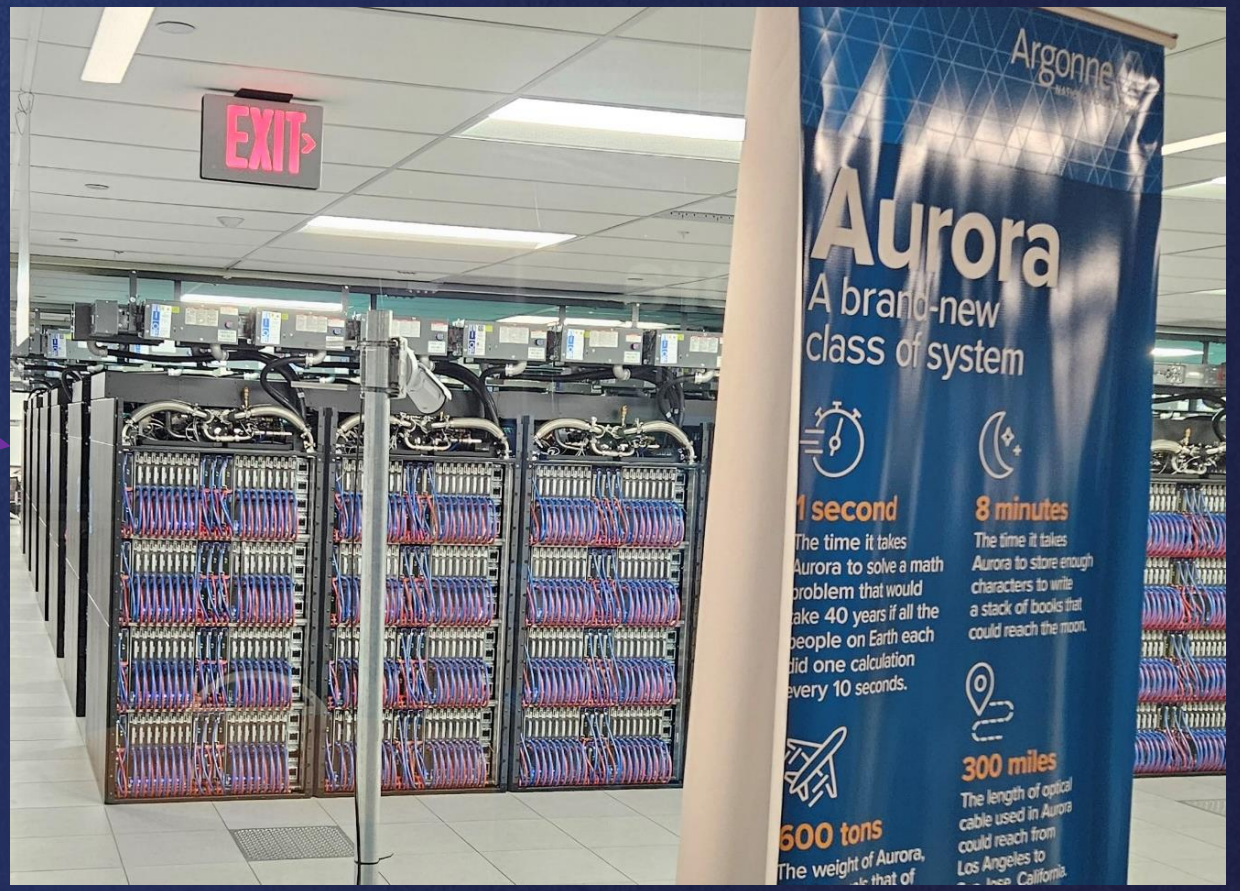
Large Scale Accelerated Rendering on 10K Ray Tracing Enabled Nodes

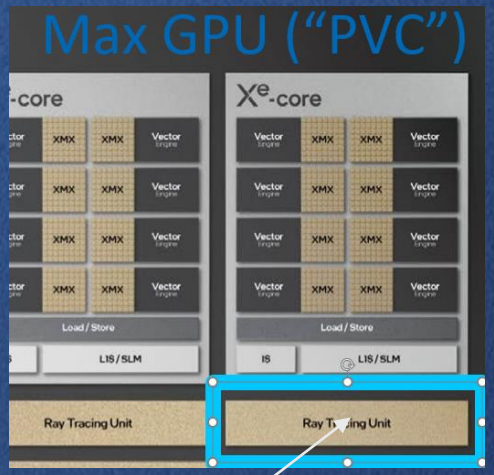


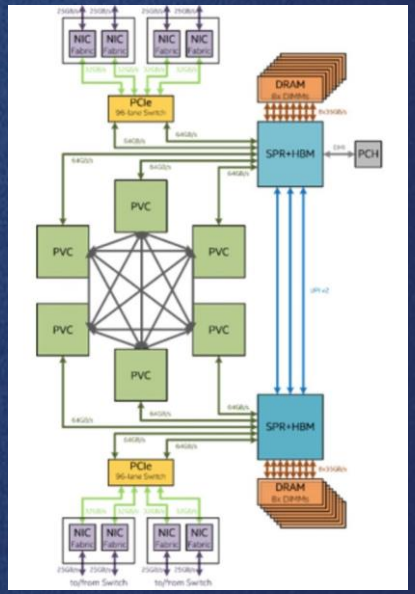
Multi-architecture, large scale workloads on Aurora



node x 10000







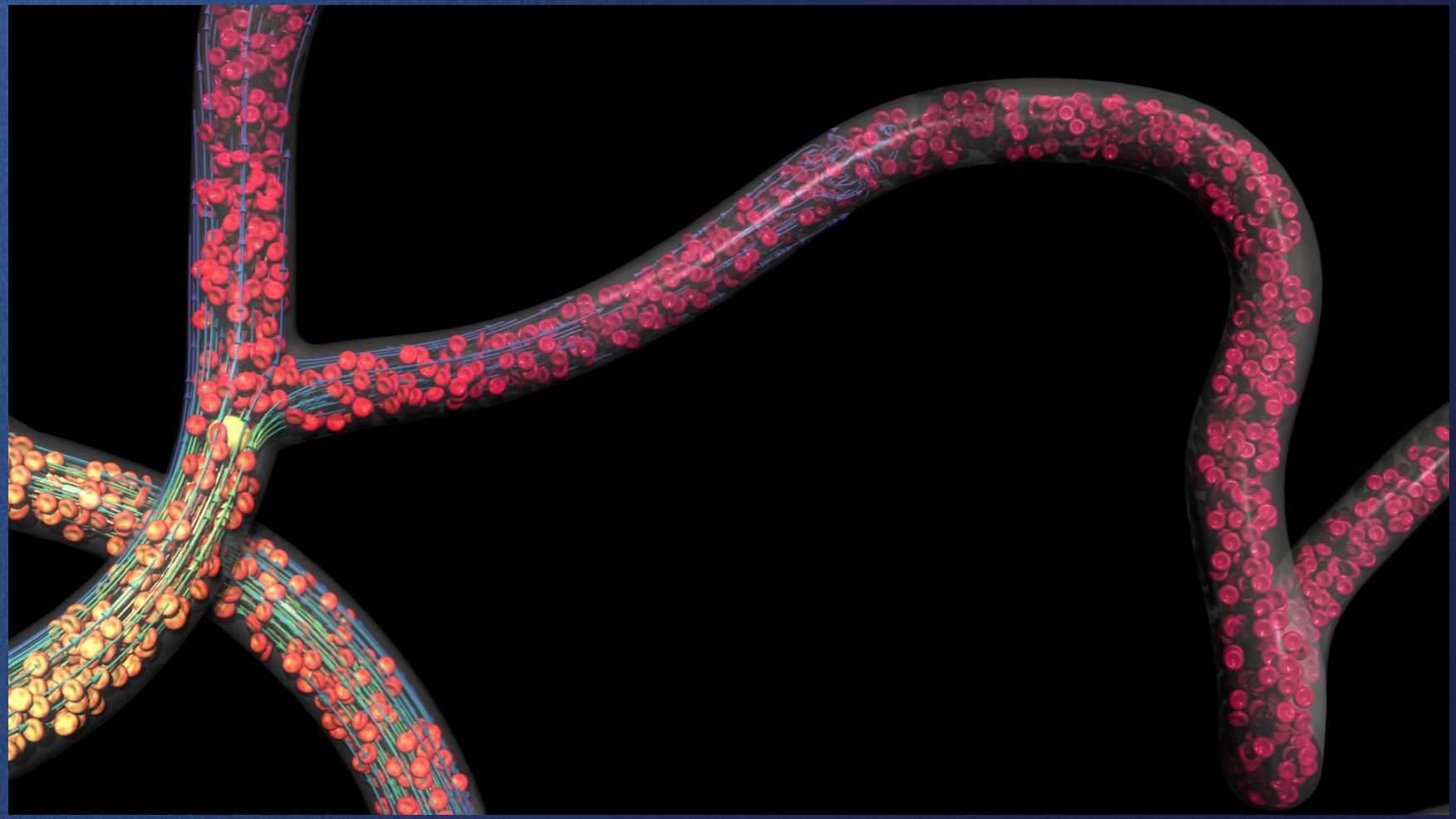
Aurora has a total of 7.68 million RTUs dedicated to ray tracing

intel

Rendering on Aurora

Argonne National Labs Aurora will compute *a lot* of numbers

Our focus – a renderer, like OpenGL, but **ray traced** and tuned for Aurora



Dataset: HARVEY, Amanda Randles, Duke University and Oak Ridge National Laboratory
Visualization: Joseph Insley, Argonne National Laboratory's Leadership Computing Facility

intel

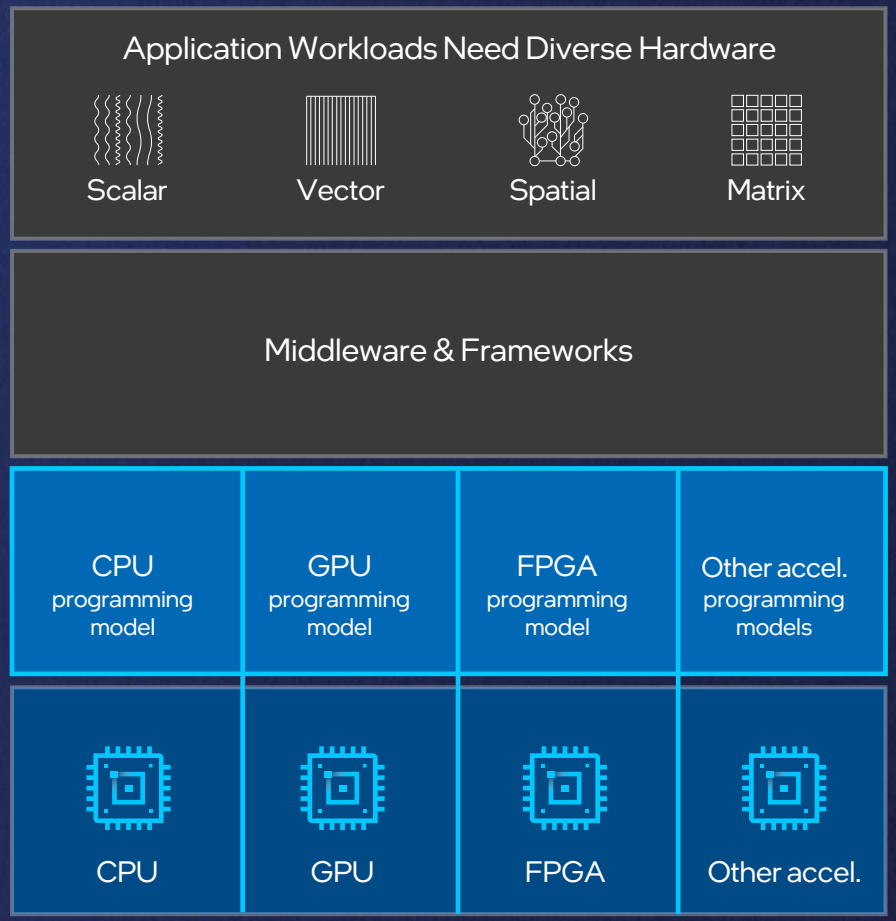
Large Scale Accelerated Rendering on 10K Ray Tracing Enabled Nodes



Programming Challenges for Multiple Architectures

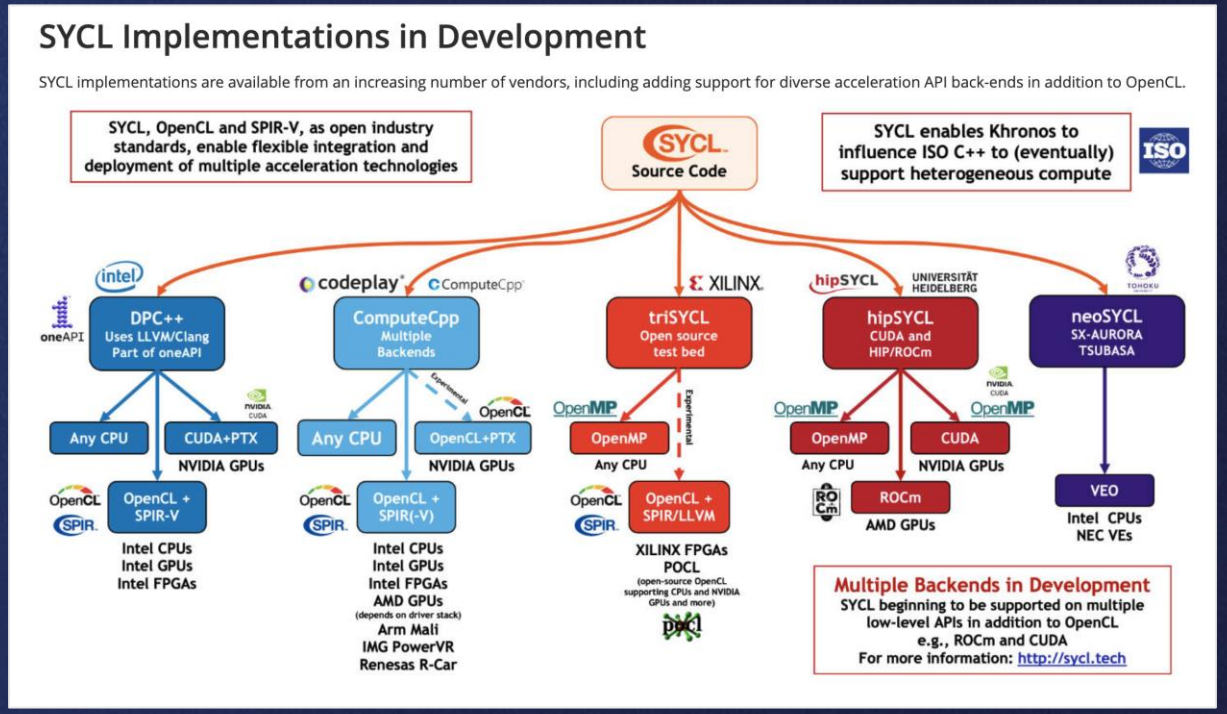
- Growth in specialized workloads
- Variety of data-centric hardware required
- Separate programming models and toolchains for each architecture are required today
- Software development complexity limits freedom of architectural choice

40% of developers target heterogeneous systems that use more than 1 type of processor, processor core, or coprocessor.



Intel® oneAPI: One Programming Model for Multiple Architectures and Vendors

- Cross-architecture code reuse for highest developer productivity
- Enables full performance of target hardware architectures for fastest time to value
- Based on open standards for freedom from proprietary language vendor lock-in
- Clear demand for vendor and architecture independent standards for heterogeneous compute
- SYCL (SYstem-wide Compute Language) is a true standard



Source: SYCL Overview - The Khronos Group Inc

Intel® oneAPI Rendering Toolkit ("RenderKit")

Intel ® OSPRay Studio
Pro-vis frontend and scene graph
https://github.com/ospray/ospray_studio

Application/In Situ aspect

ParaView/Catalyst, VisIt/libSim, /SENSEI, etc.

Intel ® OSPRay
Scalable rendering engine, API and SDK
Distributed MPI Rendering via OSPRay MPI
Intel ® implementation of the Khronos ANARI specification (https://www.khronos.org/anari)
http://www.ospray.org

Intel ® Open VKL
API for volume sampling, traversal, interpolation and classification
http://www.openvkl.org

Intel ® Embree
Optimized geometry ray tracing kernels
- BVH builders, traversal and intersection
http://www.embree.org

Intel ® Open Image Denoise
AI / DL – based denoising of sampling artifacts from path tracing
http://www.openimagedenoise.org

Scalable rendering

intel oneAPI RENDERING TOOLKIT

intel

Intel® oneAPI Rendering Toolkit ("RenderKit")

Intel ® OSPRay Studio
Pro-vis frontend and scene graph
https://github.com/ospray/ospray_studio

Application/In Situ aspect

ParaView/Catalyst, VisIt/libSim, /SENSEI, etc.

Intel ® OSPRay
Scalable rendering engine, API and SDK
Distributed MPI Rendering via OSPRay MPI
Intel ® implementation of the Khronos ANARI specification (https://www.khronos.org/anari)
http://www.ospray.org

Intel ® Open VKL
API for volume sampling, traversal, interpolation and classification
http://www.openvkl.org

Volumetric Ray Tracing



Gravational Waves :
GRChombo AMR Data, Stephen Hawking CTC, Ucambridge, Queens College, London.
Visualization: Carson Brownlee, Intel

Intel ® Embree
Optimized geometry ray tracing kernels
- BVH builders, traversal and intersection
http://www.embree.org

Geometric Ray Tracing



Peter Rabbit – Animal Logic

Intel ® Open Image Denoise
AI / DL – based denoising of sampling artifacts from path tracing
http://www.openimagedenoise.org

Denoising Filters



The Junk Shop by Alex Treviño. Original Concept by Anaïs Maamar

intel oneAPI RENDERING TOOLKIT

intel

Intel® oneAPI Rendering Toolkit ("RenderKit")

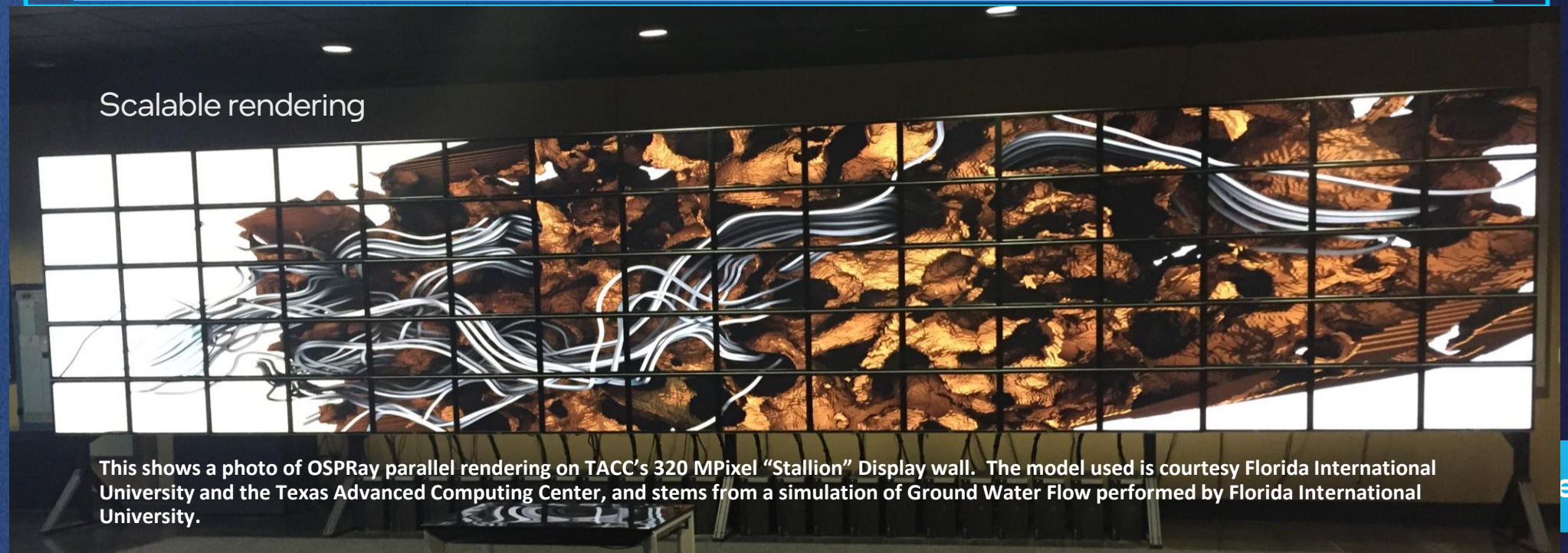
Intel ® OSPRay Studio
Pro-vis frontend and scene graph
https://github.com/ospray/ospray_studio

Application/In Situ aspect

ParaView/Catalyst, VisIt/libSim, /SENSEI, etc.

Intel ® OSPRay
Scalable rendering engine, API and SDK
Distributed MPI Rendering via OSPRay MPI
Intel ® implementation of the Khronos ANARI specification (https://www.khronos.org/anari)
http://www.ospray.org

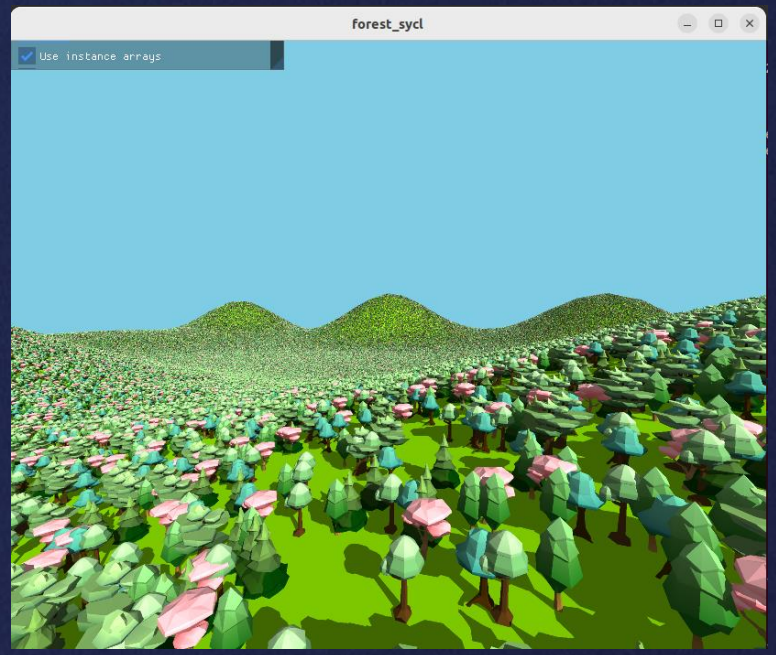
intel
oneAPI
RENDERING TOOLKIT



This shows a photo of OSPRay parallel rendering on TACC's 320 MPixel "Stallion" Display wall. The model used is courtesy Florida International University and the Texas Advanced Computing Center, and stems from a simulation of Ground Water Flow performed by Florida International University.

Intel® Embree Updates starting with v4.0

- Support for Intel® Arc™ GPUs (DG2), Data Center GPU Flex Series (ATS-m), and Data Center GPU Max Series (PVC)
- Ray tracing hardware acceleration enabled
- Unified host-side code
  - API consistency
  - Minimizes effort to port to GPU
- Specialization constants and feature flags → reduce kernel size and improve performance
- Instance arrays primitive for reducing memory requirements

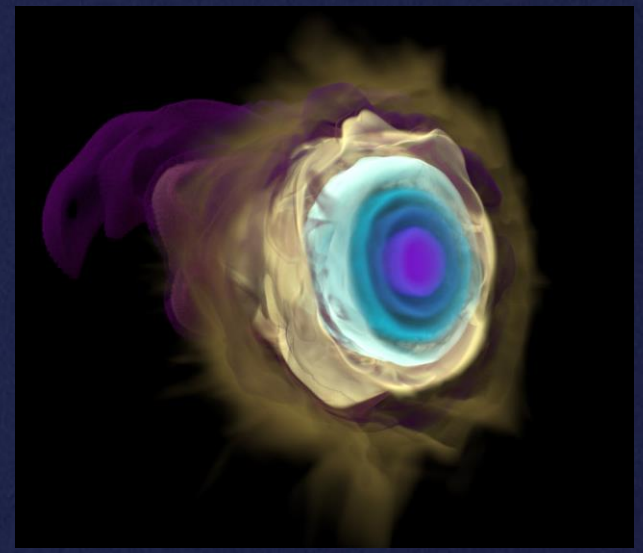


Screenshot of Embree forest example.

Intel® OpenVKL Updates in v2.0

- Support for Intel® Arc™ GPUs (DG2), Data Center GPU Flex Series (ATS-m), and Data Center GPU Max Series (PVC)
- Unified host-side code
  - API consistency
  - Minimizes effort to port to GPU
- Specialization constants and feature flags → reduce kernel size and improve performance
- Supports zero-copy for structured data → in situ



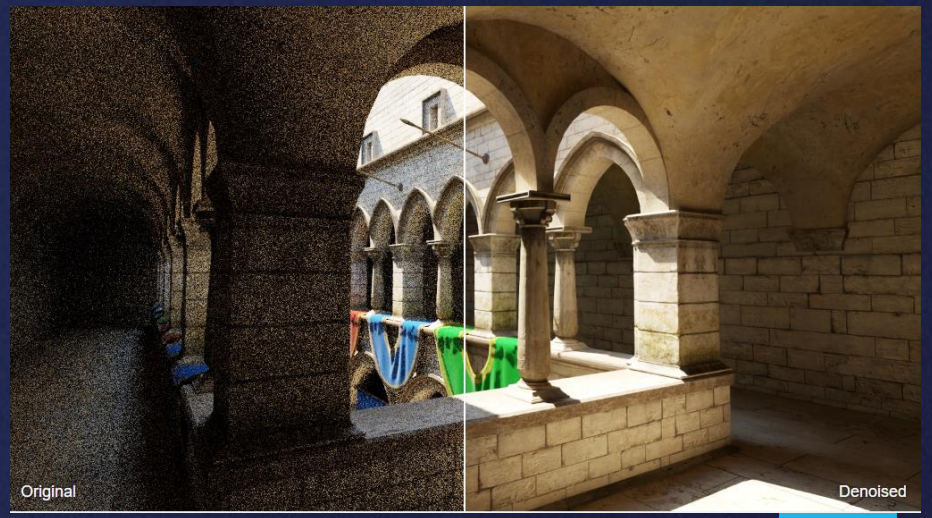


Visualization by Greg P. Johnson, Intel, and Joseph Insley, Argonne National Lab. Simulation provided by the University of California, Santa Barbara (team leads Yan-Fei Jiang and Lars Bildsten).

Intel® Open Image Denoise starting with v2.0

- Support for
  - Intel® GPUs → SYCL
  - Nvidia® GPUs → CUDA
  - AMD® GPUs → HIP
- Compatibility with any kind of device via OIDNBuffer
- Asynchronous read and write on GPU
- Asynchronous operation execution on GPU
- Same image quality as on CPUs, added filter quality modes: "high" (default), "balanced" (faster)





Scene courtesy of Frank Meinl, downloaded from Morgan McGuire's Computer Graphics Archive.

intel

Intel® OSPRay Updates in v3.0

- Support for Intel® Arc™ GPUs (DG2), Data Center GPU Flex Series (ATS-m), and Data Center GPU Max Series (PVC)
- Specialization constants and feature flags → reduce kernel size and improve performance → pass specialization constants to Embree & OpenVKL
- GPU-optimized volume raymarchers for SciVis and AO renderers → performance improvement
- Sparse framebuffer for *parallel modules*
  - Uses a tile-based work decomposition among MPI ranks or multidevices
  - Rendering work for a tile is decomposed into rendering tasks

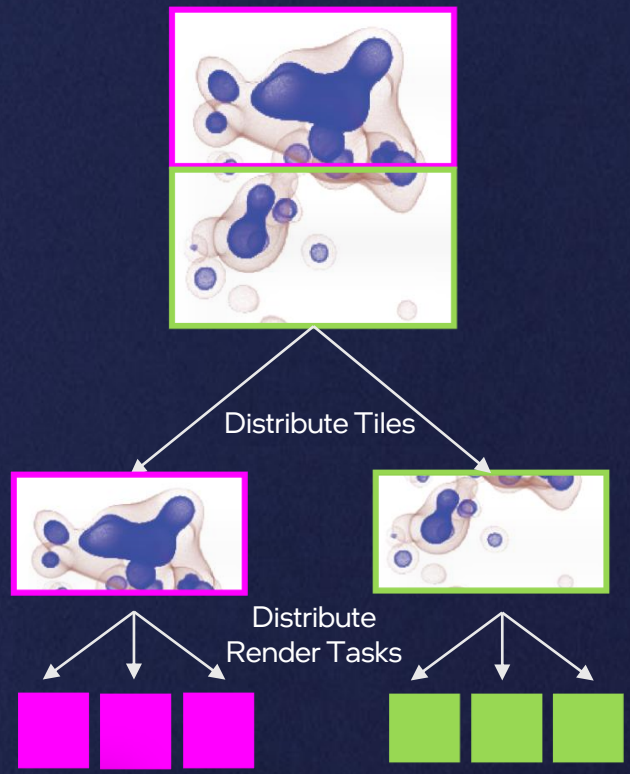
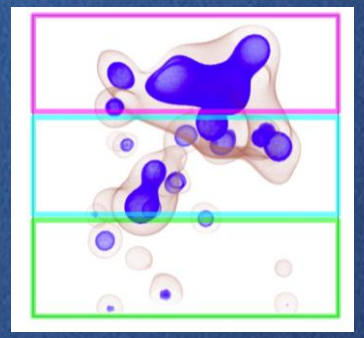


Image Parallel

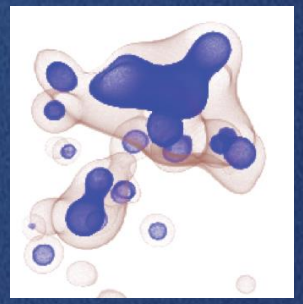
Distribute image tiles/pixels



Tiles Distribution

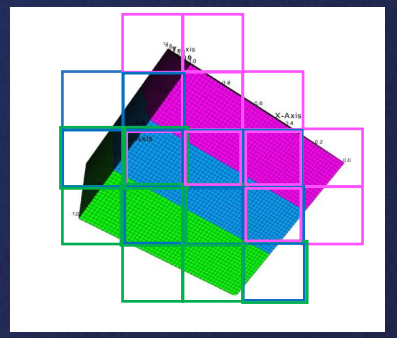
Render Tile Render Tile Render Tile

Communication



Data Parallel

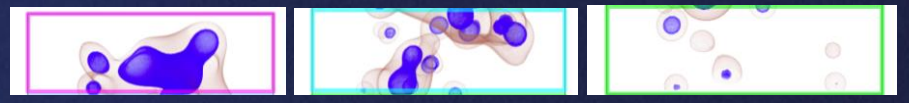
Distribute data blocks



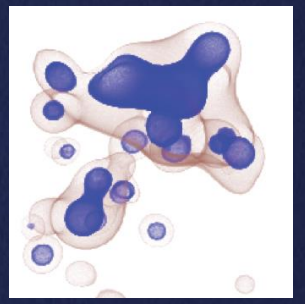
Data Distribution

Render Data Render Data Render Data

Communication



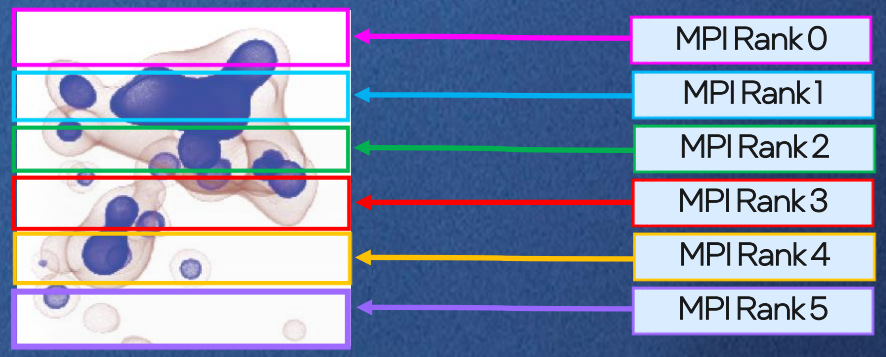
Communication



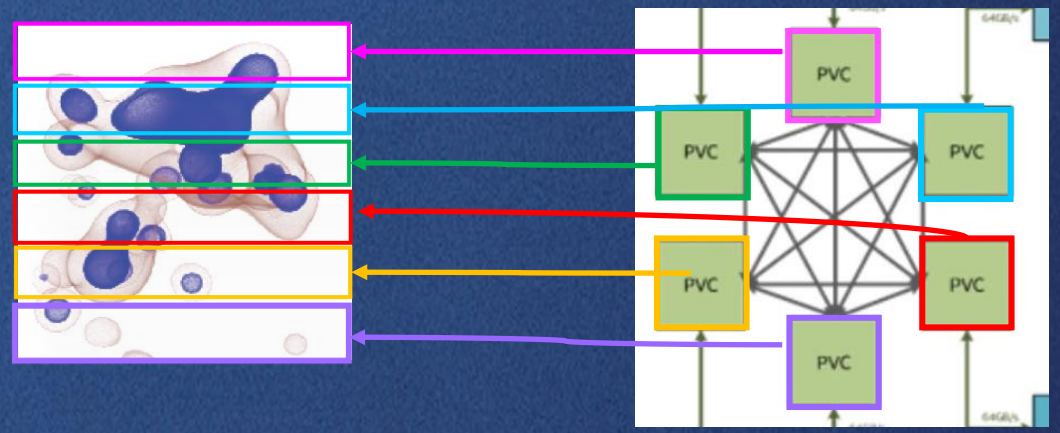
intel

Image Parallel

MPI-Offload: Image Parallel between nodes

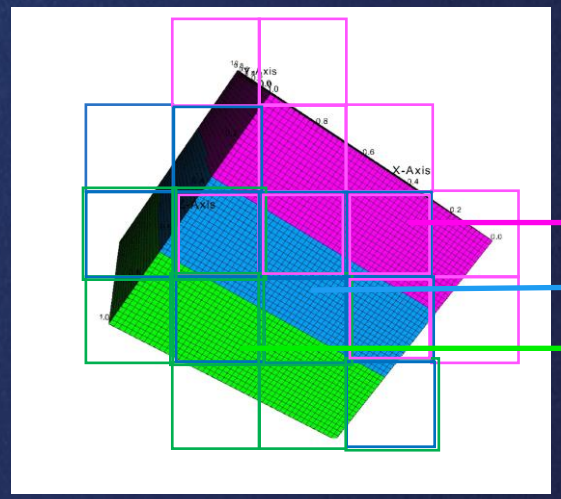


Multi-Device: Image Parallel within a node



Data Parallel

MPI-Distributed: Data Parallel between nodes



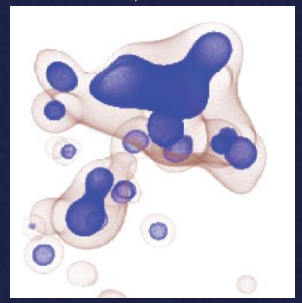
OSPRay Studio

OSPRay Studio

OSPRay Studio

← MPICH →

Distributed Frame Buffer



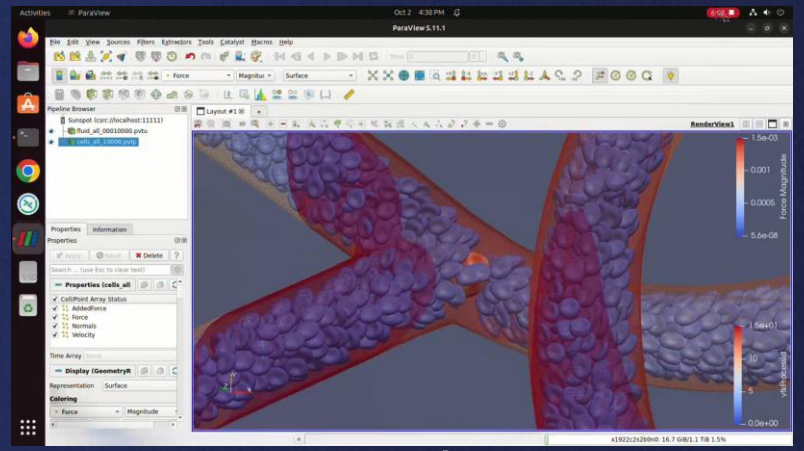
- ✓ Usher, Will, et al. "Scalable ray tracing using the distributed framebuffer." Computer Graphics Forum. Vol. 38. No. 3. 2019.
- ✓ Moreland, Kenneth D. *IceT users' guide and reference*. No. SAND2010-7451. Sandia National Laboratories (SNL), Albuquerque, NM, and Livermore, CA (United States), 2011.

intel

Third Party Applications Running using Intel® OSPRay 3

- v3 API minimally changed from v2
- Demonstrated compatibility with VTK based visualization frameworks on Argonne's Aurora Supercomputer
- Downstream MRs in progress

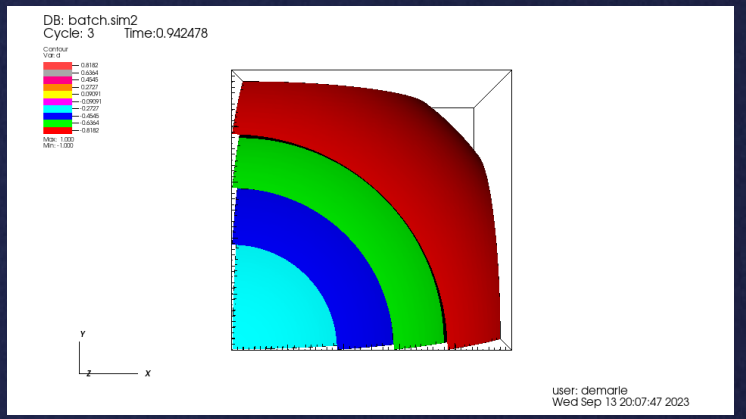
- NCAR Vapor and Ansys EnSight soon...



*ParaView/Catalyst

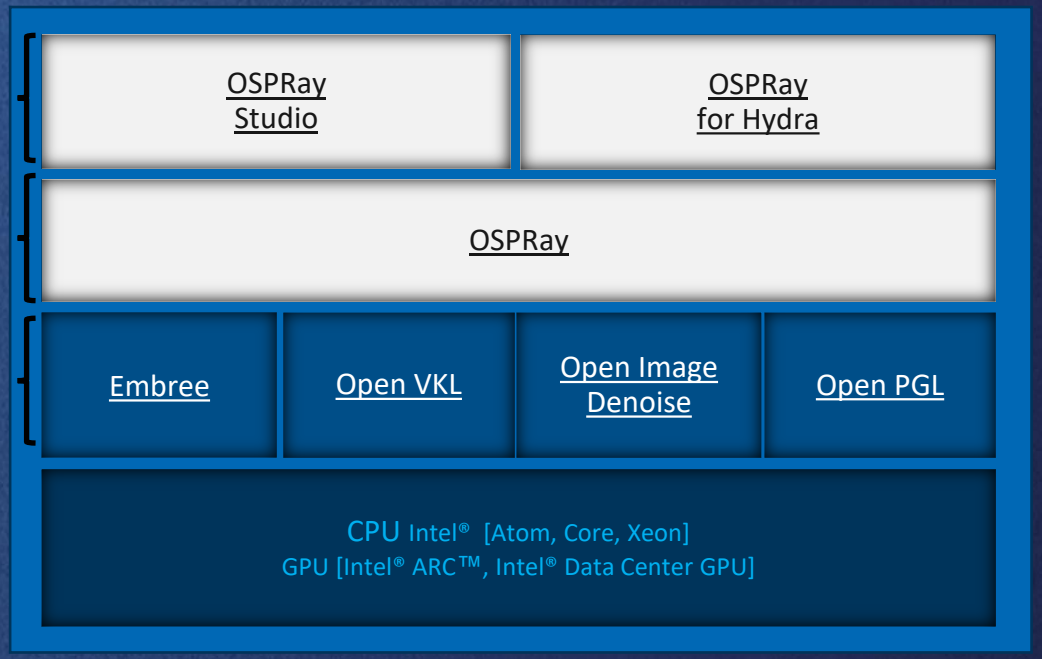


*SENSEI



*VisIt/libSim

RenderKit for all types of Viz
Cross Platform – Cross Architecture



Access to the OSPRay renderer by way of a **GUI Application** or **Plug-in to enable OpenUSD workflows**

Renderer

Embree **Geometric Ray Tracing,**
Open VKL **Volumetric Ray Tracing,**
Open Image Denoise **Denoising,**
Open PGL **Path Guiding**
Capabilities can be integrated into any renderer to yield high fidelity rendered images at high quality and increased performance

GPU Ray Tracing HW Acceleration



Rendering ToolKit enables Viz at Scale

- Digital Twins → a natural fit large scale Viz problems
- Large problem scope → a one size fits all solution won't work
- Cross Platform → Cross Architecture gives flexibility to address different problems
- Single code base provides stability



Intel® oneAPI Rendering ToolKit

No Transistor left behind !

No Developer left behind !

No Pixel left behind !



Model: Bentley Motors
Visualization: Bruce Cherniak, Intel

Visualizing Tumor Cell Trajectories on Ray Tracing Enabled Nodes is on the demo floor now! Booth #: 617



intel

Legal Notices and Disclaimers

- Performance varies by use, configuration and other factors. Learn more at www.Intel.com/PerformanceIndex.
- Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. See backup for configuration details. No product or component can be absolutely secure.
- Your costs and results may vary.
- Intel technologies may require enabled hardware, software or service activation.
- © Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. * Other names and brands may be claimed as the property of others.
- Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

it starts with intel

Thank you!

roba.binyahib@intel.com
david.demarle@intel.com
krishnan.ramachandran@intel.com