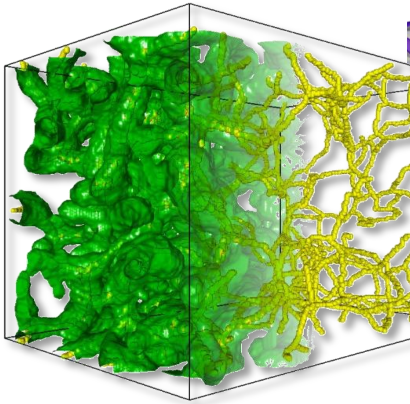
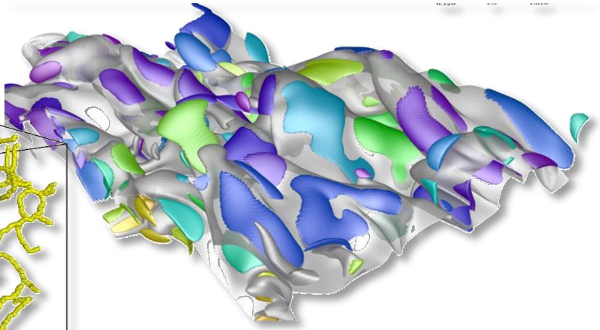
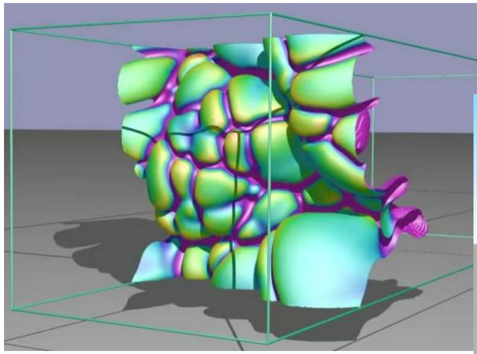
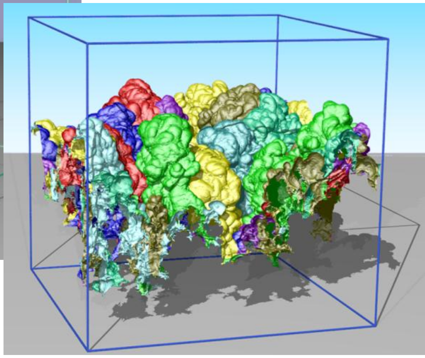
CEDMAV Center for Extreme Data Management, Analysis, and Visualization

Management, Analysis and Visualization of Massive Scientific Data

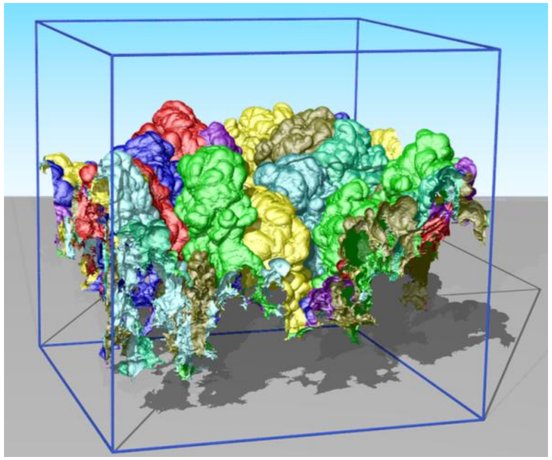





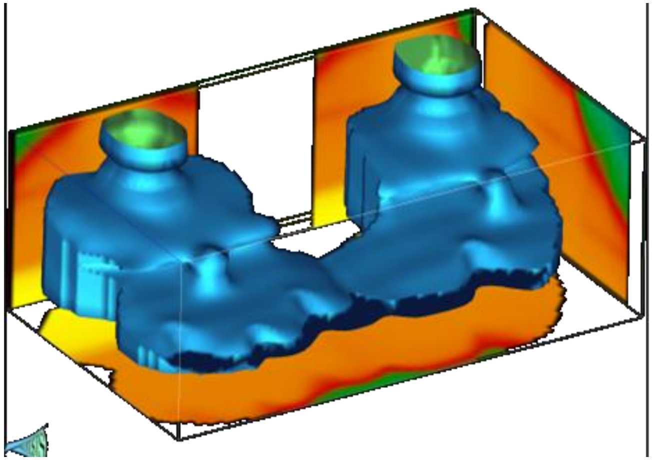
Peer-Timo Bremer

Valerio Pascucci

Center for Extreme Data Management Analysis and Visualization

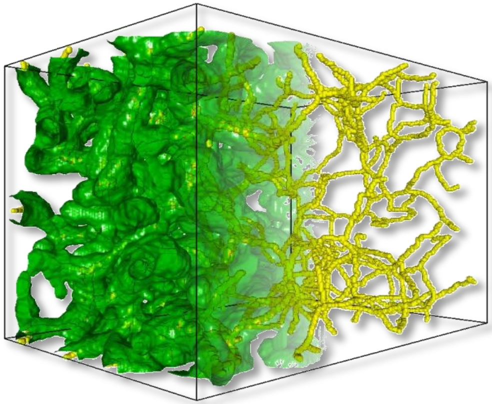
Center for Extreme Data Management, Analysis and Visualization





CEDMAV Mission

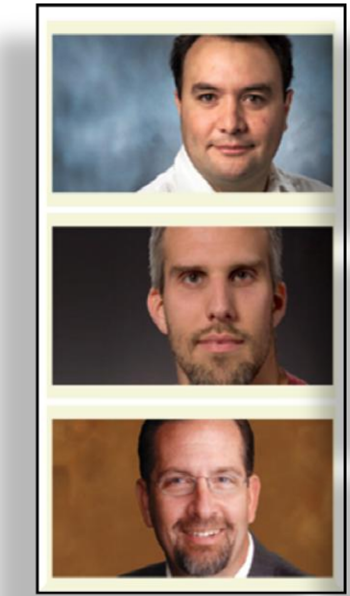
Research Future Technologies for Knowledge Extraction from Extreme Sized Data

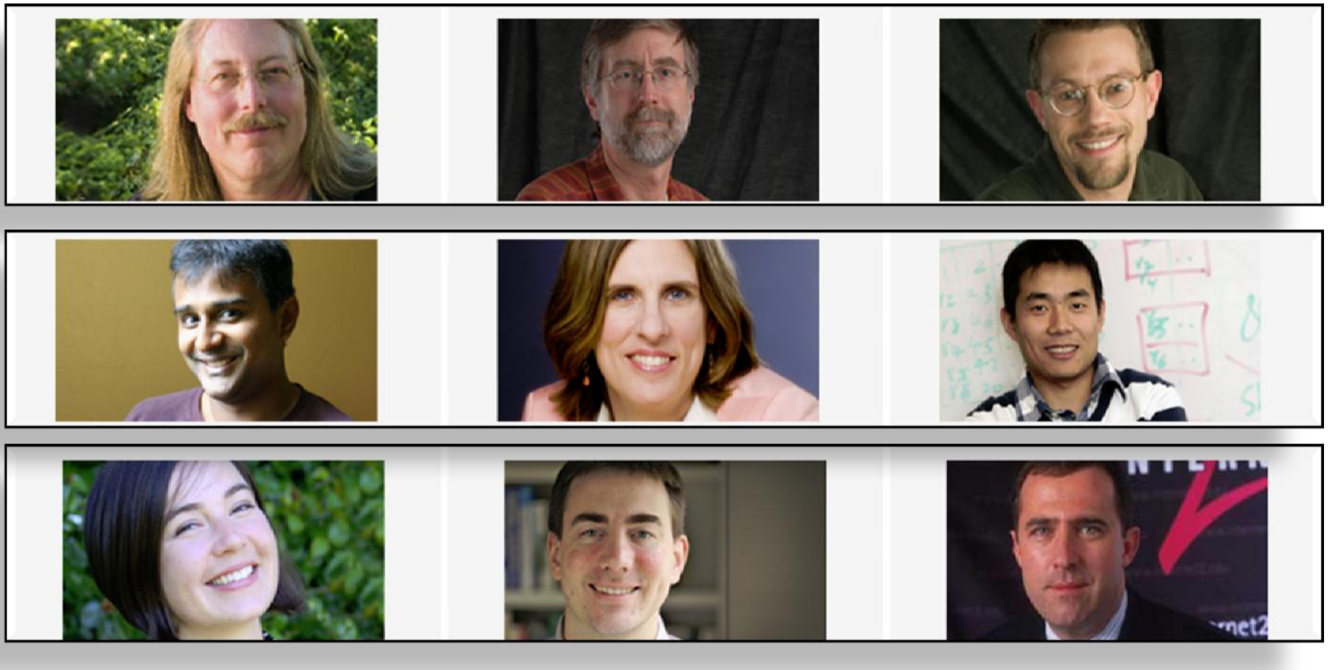


Deployment and Application of State of the Art Tools in Data Intensive Science Discovery

Education of the Next Generation Workforce Supporting Data Intensive Science and Engineering

CEDMAV





- 10 Faculty + scientists, developers, students, …
- Other partnerships: NSA, Battelle, ….
- Involvement in national Initiatives:

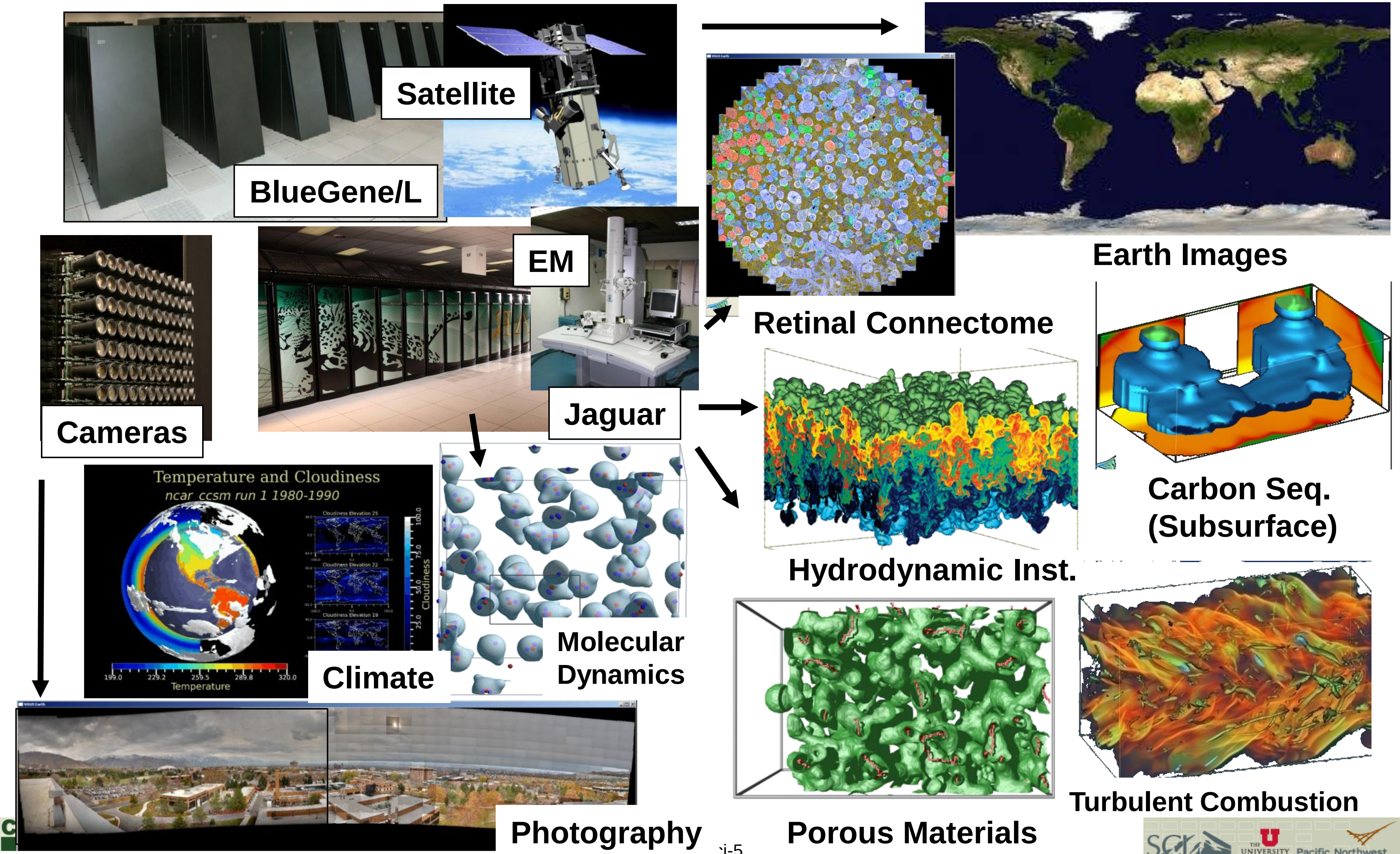
SDAV Scalable Data Management, Analysis, and Visualization

ExaCT

SUPER

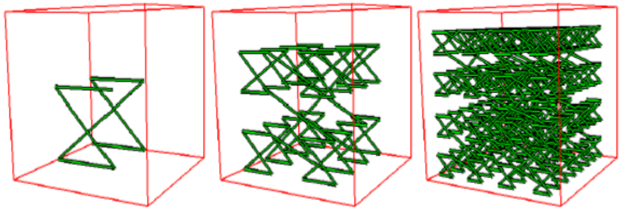
$1.6B NSA data center
(1.5 million-square-foot facility)

Massive Simulation and Sensing Devices Generate Great Challenges and Opportunities

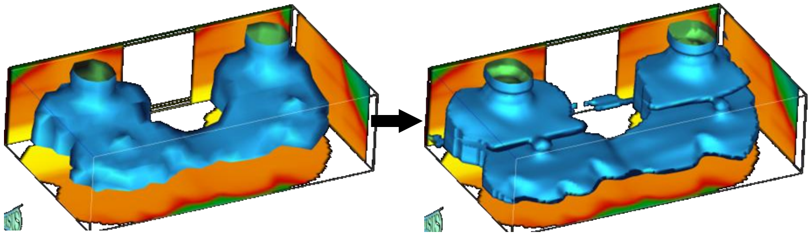


A Science Cyberinfrastructure Requires Efficient Big Data Management and Processing

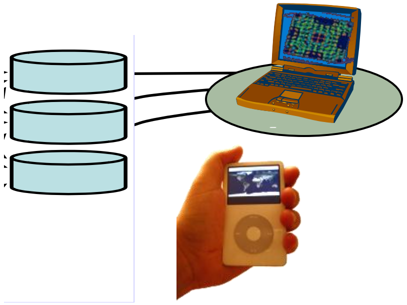
- Advanced data storage techniques:
  - Data re-organization.
  - Compression.



- Advanced algorithmic techniques:
  - Streaming.
  - Progressive multi-resolution.
  - Out of core computations.

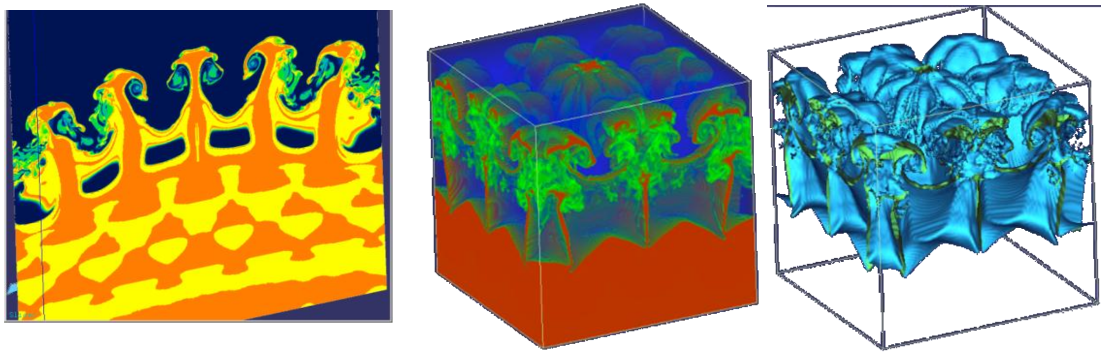


- Scalability across a wide range of running conditions:
  - From laptop, to office desktop, to cluster of PC, to BG/L.
  - Memory, to disk, to remote data access.

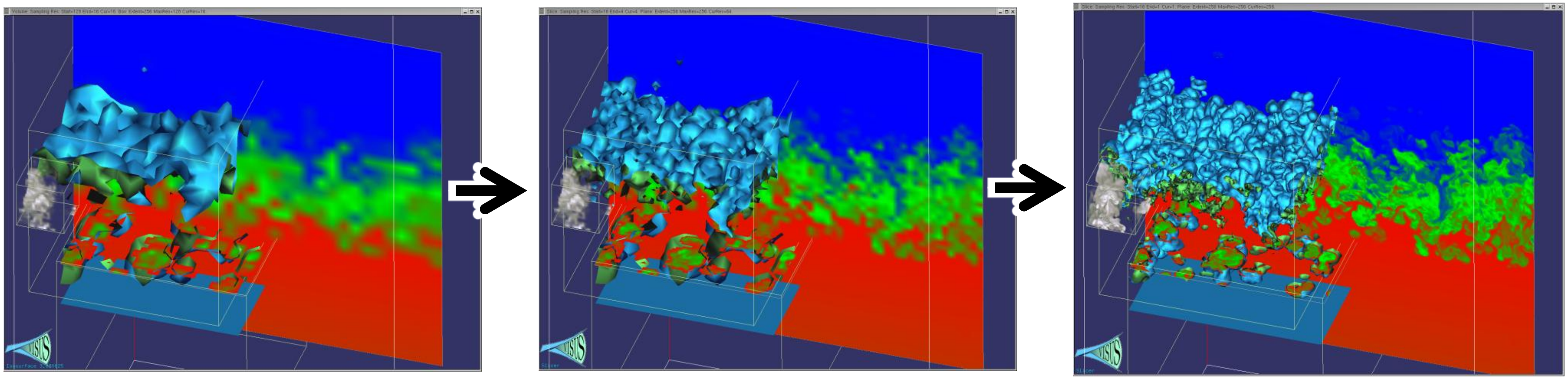


We Redesigned the Data Management and Visualization Pipeline with New Principles

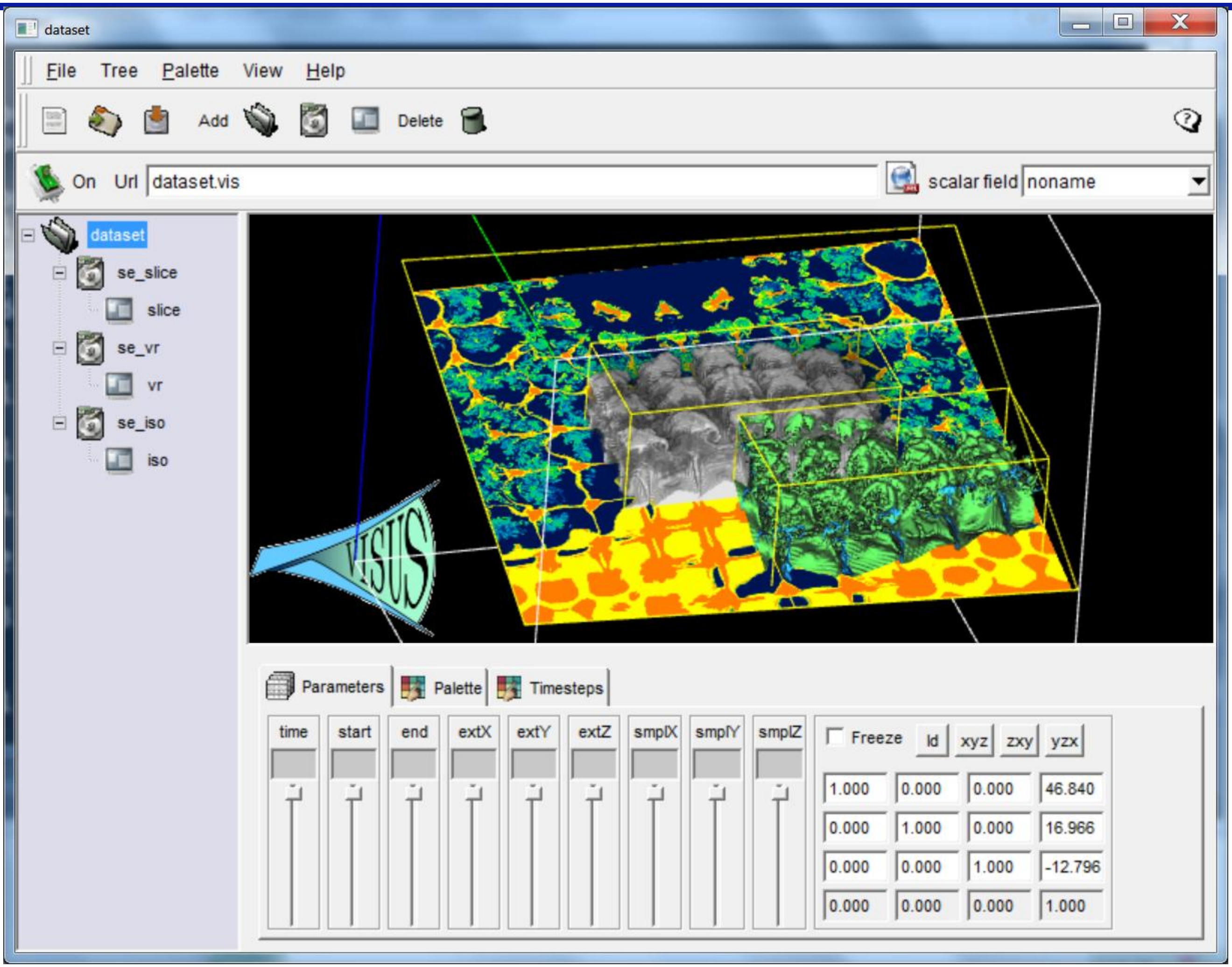
- Basic core techniques:
  - Slicing, Volume rendering, Iso-surfaces
  - Topology
  - Statistics



- Cache-oblivious out-of-core processing optimizing access locality for any size of data blocks
- Pipelines of progressive algorithms
- Coarse-to-fine construction of multi-resolution models
- Remote data streaming



Basic Demo



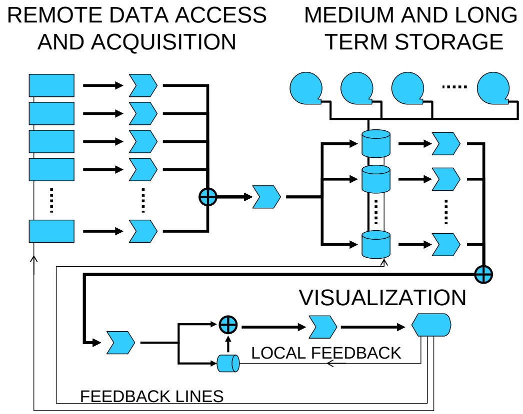
Url: dataset.vis

scalar field: noname

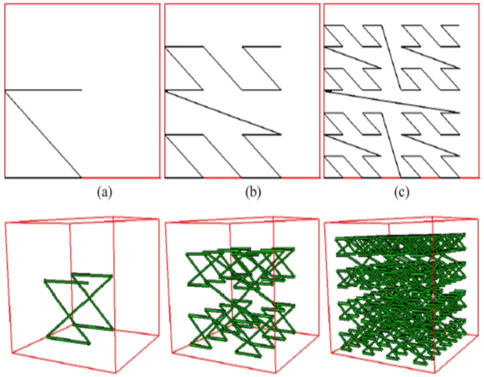
☐ Freeze

1.000	0.000	0.000	46.840
0.000	1.000	0.000	16.966
0.000	0.000	1.000	-12.796
0.000	0.000	0.000	1.000

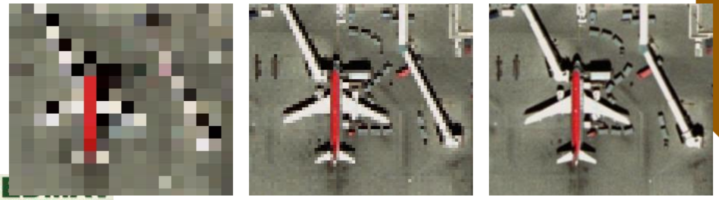
We Consider the Three Main Components Defining a Computing Infrastructure



Processing Network
(Data Access Path)

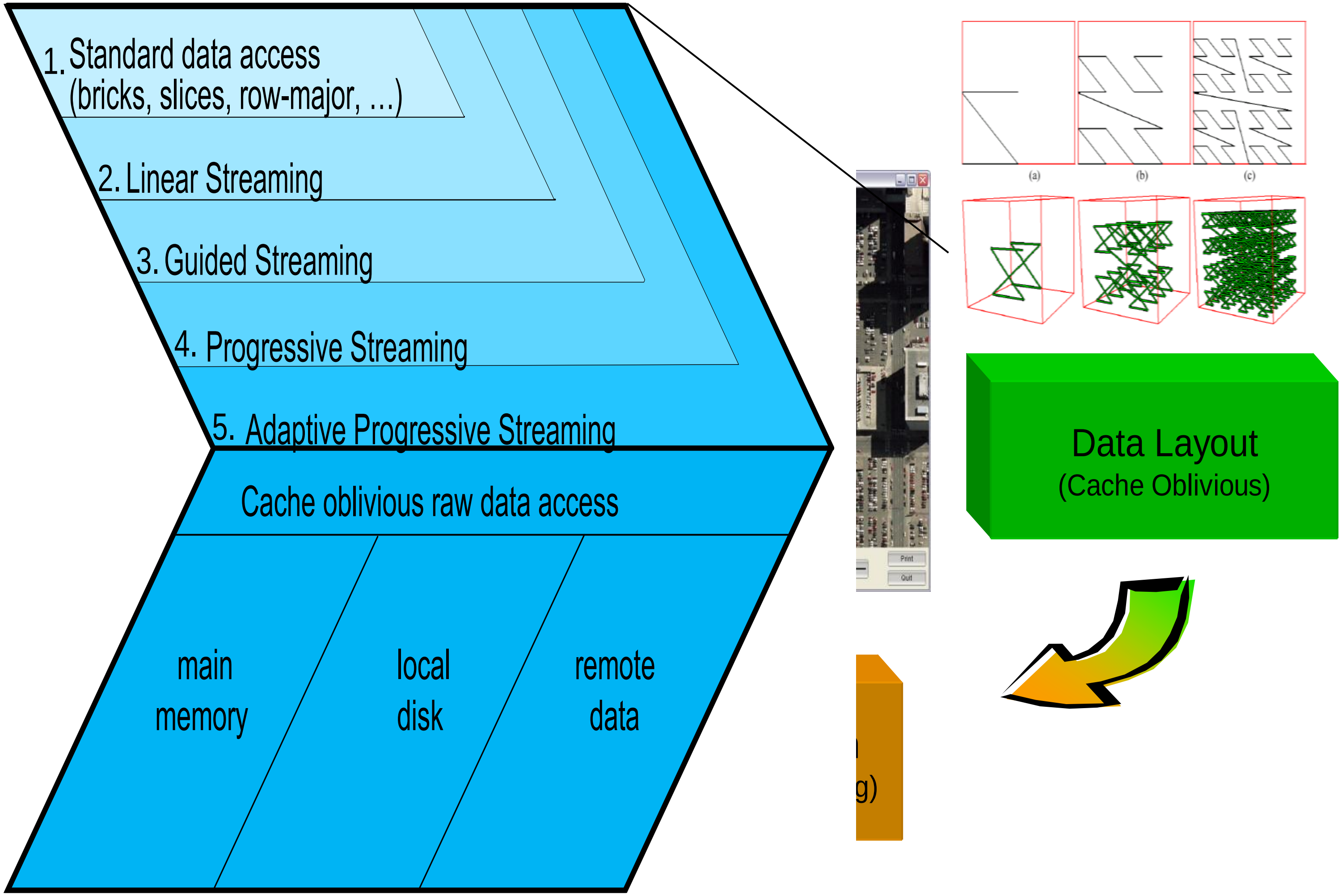


Data Layout
(Cache Oblivious)



Algorithm Design
(Progressive Processing)

We Characterize Algorithmic Classes Based on Effect in a Processing Network



1. Standard data access (bricks, slices, row-major, ...)
2. Linear Streaming
3. Guided Streaming
4. Progressive Streaming
5. Adaptive Progressive Streaming

Cache oblivious raw data access

main memory

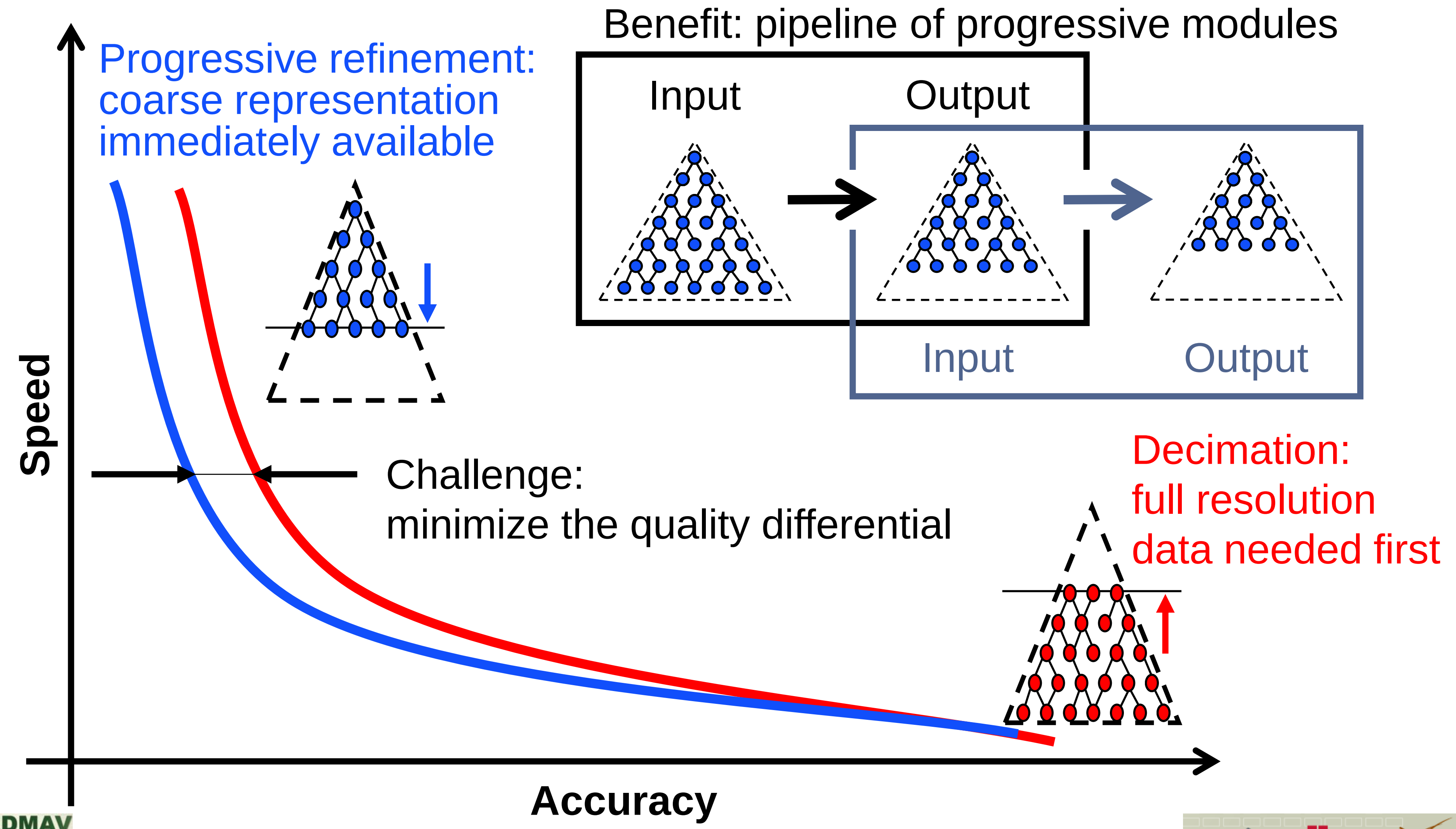
local disk

remote data

Data Layout

(Cache Oblivious)

The use of top-down and bottom-up processes have a strong impact on the data stream



Benefit: pipeline of progressive modules

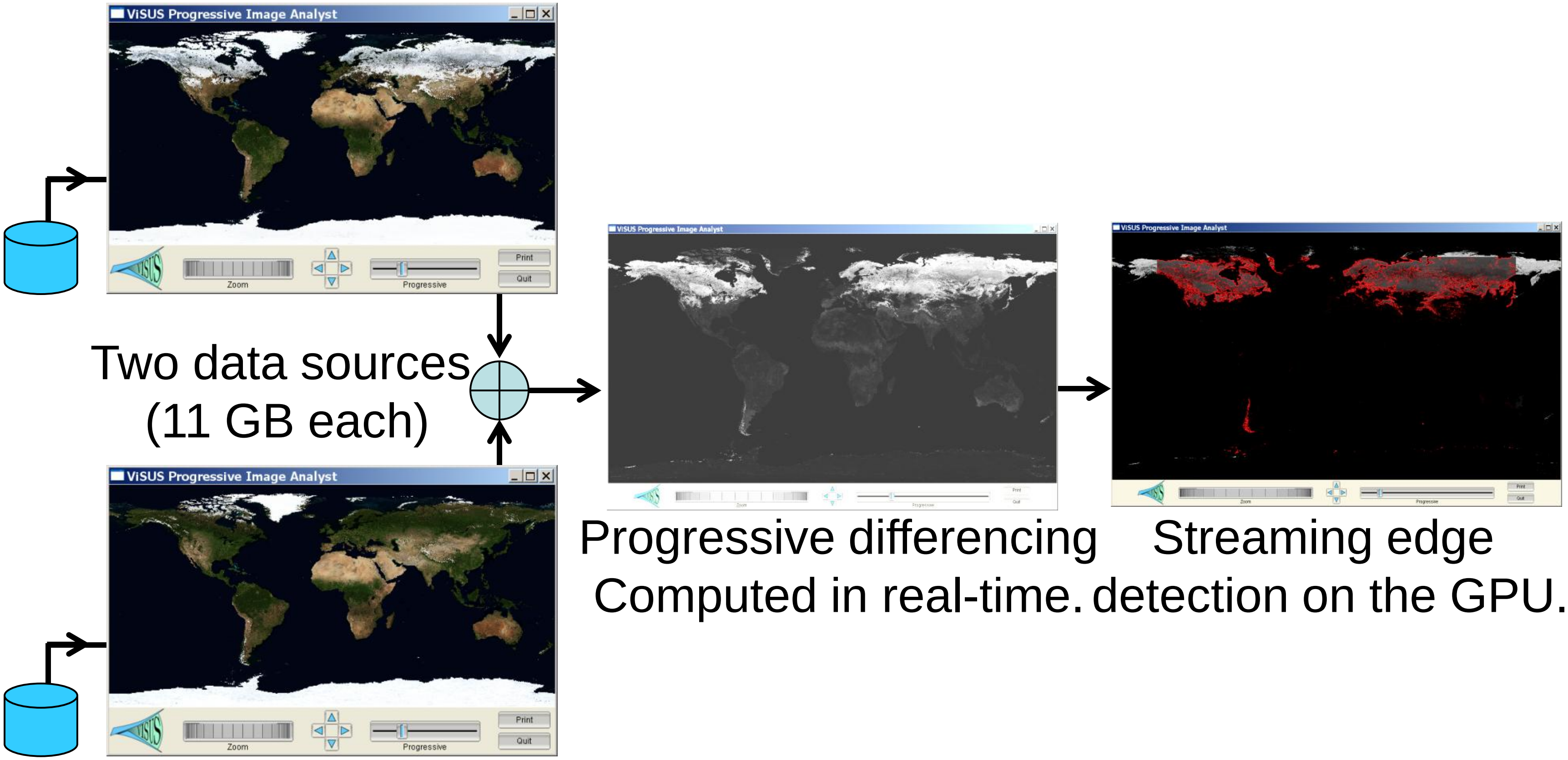
Progressive refinement: coarse representation immediately available

Challenge:
minimize the quality differential

Decimation:
full resolution
data needed first

We Allow Distributed Computations at Different Stages of the Data Stream

- Progressive Image Differencing + Editable GPU filter.



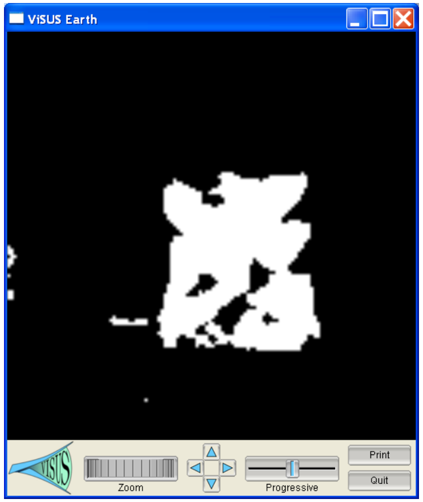
Two data sources
(11 GB each)

Progressive differencing
Computed in real-time.

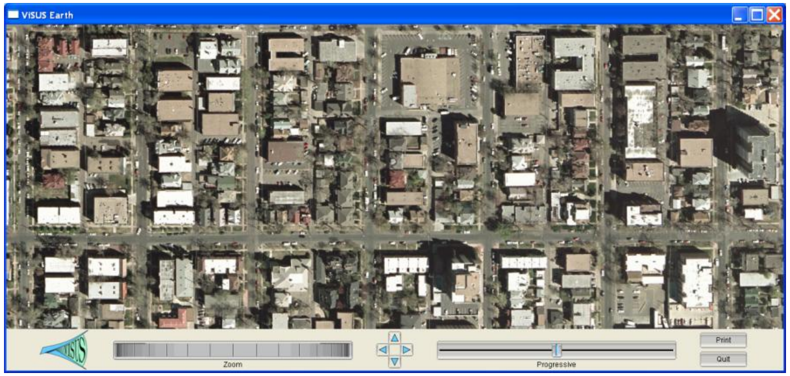
Streaming edge
detection on the GPU.

We are Developing Progressive Scheme for Content Based Image Processing

- **Hypothesis:**



- **Progressive Analysis:**

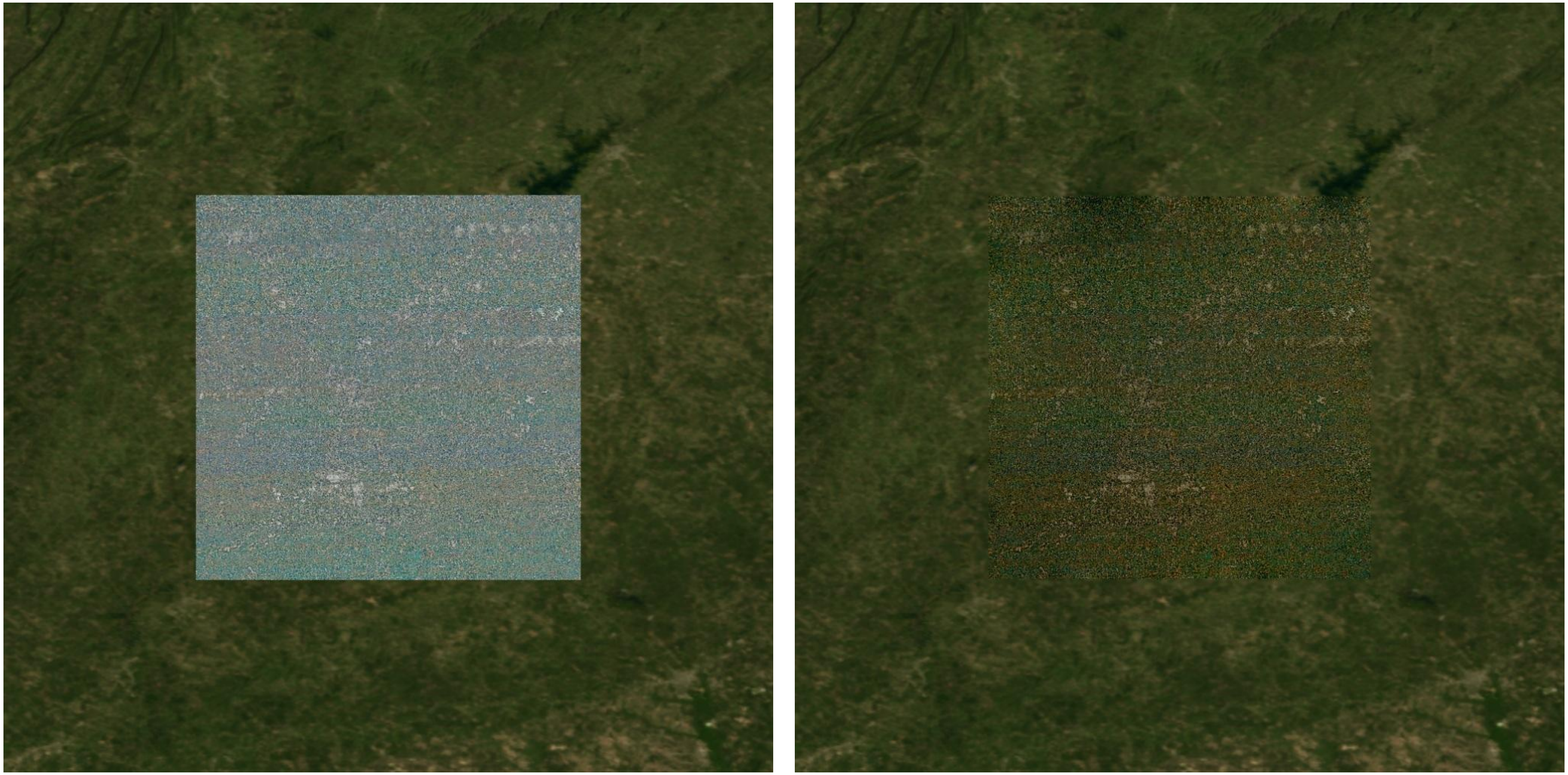


Poisson Solver for Image Cloning in Massive Image Collections

- **Color correction of 600+ images in real time**

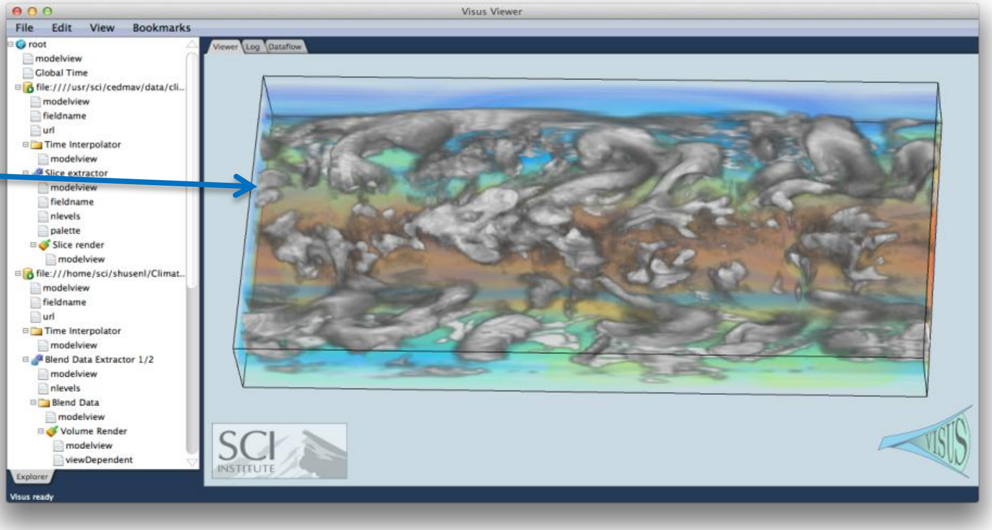
Poisson Solver for Image Cloning in Massive Image Collections

- **Pasting a 300GB satellite image of a city in background world map merged in real time**

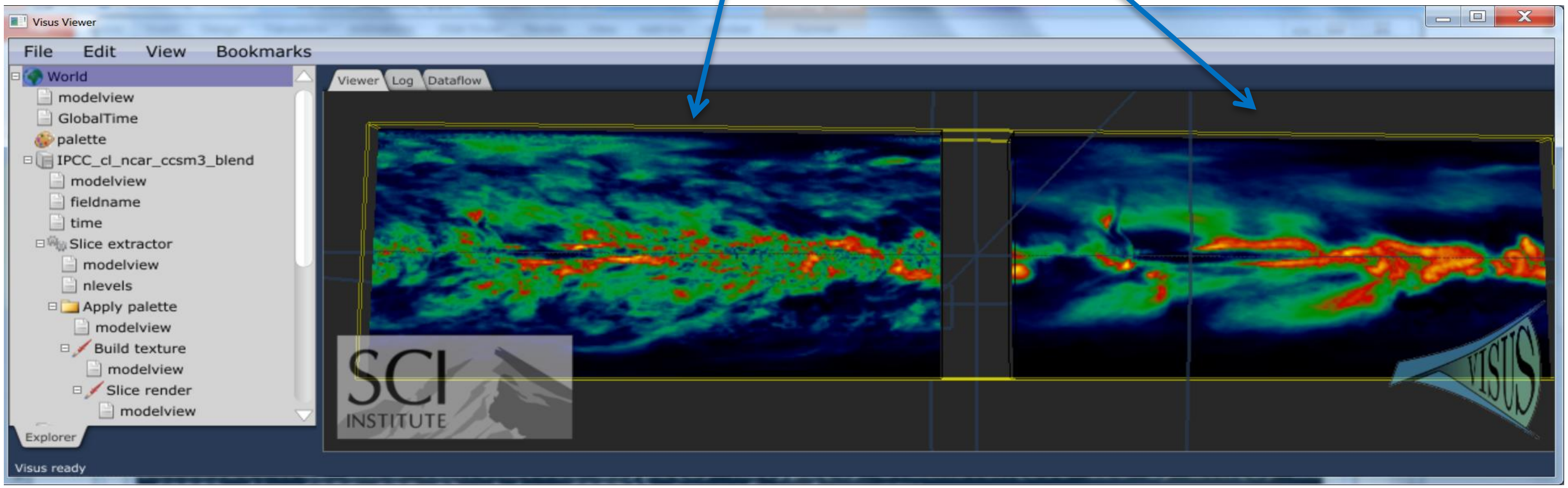


ViSUS Remote climate Data Analysis and Visualization

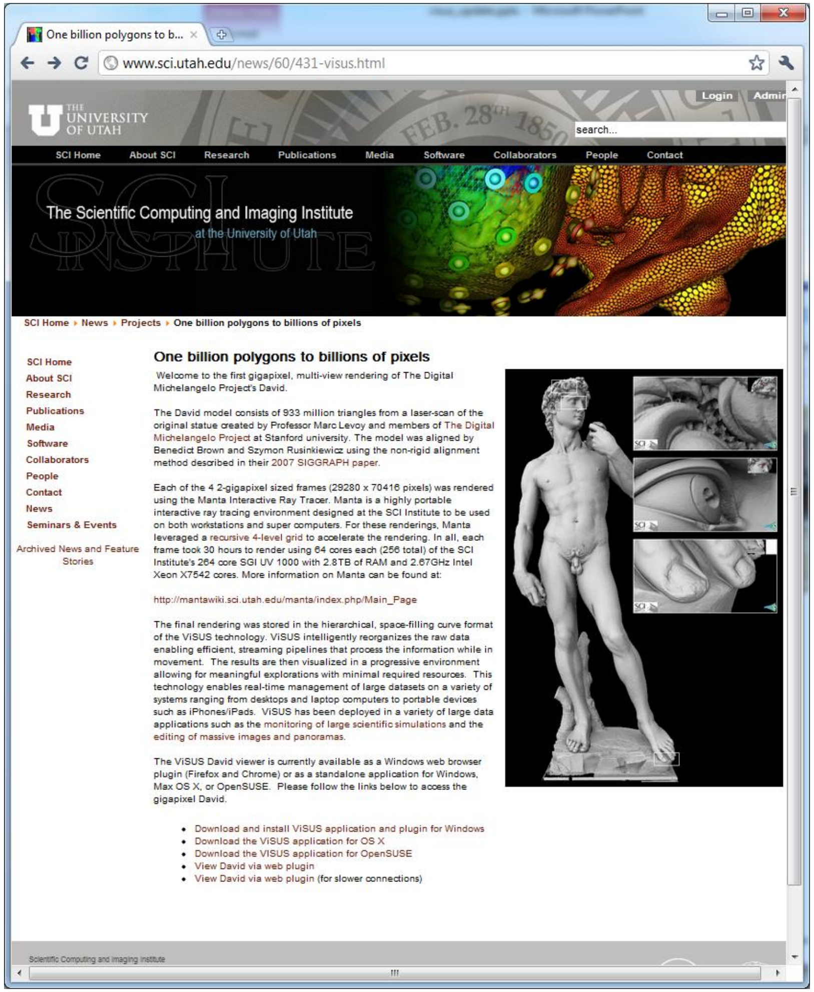
- ViSUS data streams allow to merging multiple datasets in real time
- Time interpolation of and concurrent visualization of climate data ensembles defined on different time scales
- Server side and client side computation of statistical functions such as median, average, standard deviation, …….



Standard Deviation and Average of ten climate models



Server can be wrapped in Apache plug-in
Client can be run in a web browser



SCI Home ▸ News ▸ Projects ▸ One billion polygons to billions of pixels

- SCI Home
- About SCI
- Research
- Publications
- Media
- Software
- Collaborators
- People
- Contact
- News
- Seminars & Events

Archived News and Feature Stories

One billion polygons to billions of pixels

Welcome to the first gigapixel, multi-view rendering of The Digital Michelangelo Project's David.

The David model consists of 933 million triangles from a laser-scan of the original statue created by Professor Marc Levoy and members of The Digital Michelangelo Project at Stanford university. The model was aligned by Benedict Brown and Szymon Rusinkiewicz using the non-rigid alignment method described in their 2007 SIGGRAPH paper.

Each of the 4 2-gigapixel sized frames (29280 x 70416 pixels) was rendered using the Manta Interactive Ray Tracer. Manta is a highly portable interactive ray tracing environment designed at the SCI Institute to be used on both workstations and super computers. For these renderings, Manta leveraged a recursive 4-level grid to accelerate the rendering. In all, each frame took 30 hours to render using 64 cores each (256 total) of the SCI Institute's 264 core SGI UV 1000 with 2.8TB of RAM and 2.67GHz Intel Xeon X7542 cores. More information on Manta can be found at:

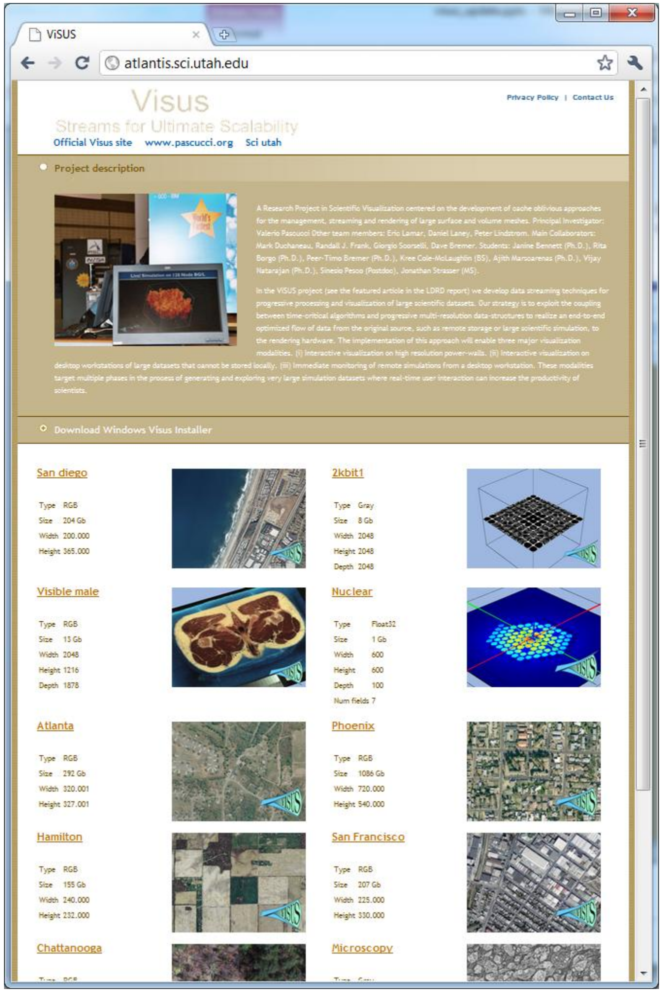
http://mantawiki.sci.utah.edu/manta/index.php/Main_Page

The final rendering was stored in the hierarchical, space-filling curve format of the ViSUS technology. ViSUS intelligently reorganizes the raw data enabling efficient, streaming pipelines that process the information while in movement. The results are then visualized in a progressive environment allowing for meaningful explorations with minimal required resources. This technology enables real-time management of large datasets on a variety of systems ranging from desktops and laptop computers to portable devices such as iPhones/iPads. ViSUS has been deployed in a variety of large data applications such as the monitoring of large scientific simulations and the editing of massive images and panoramas.

The ViSUS David viewer is currently available as a Windows web browser plugin (Firefox and Chrome) or as a standalone application for Windows, Max OS X, or OpenSUSE. Please follow the links below to access the gigapixel David.

- Download and install ViSUS application and plugin for Windows
- Download the ViSUS application for OS X
- Download the VISUS application for OpenSUSE
- View David via web plugin
- View David via web plugin (for slower connections)

Scientific Computing and Imaging Institute



Project description

A Research Project in Scientific Visualization centered on the development of cache oblivious approaches for the management, streaming and rendering of large surface and volume meshes. Principal Investigator: Valerio Pascucci Other team members: Eric Lamar, Daniel Laney, Peter Lindstrom. Main Collaborators: Mark Duchaneau, Randall J. Frank, Giorgio Scorselli, Dave Bremer. Students: Janine Bennett (Ph.D.), Rita Borgo (Ph.D.), Peer-Timo Bremer (Ph.D.), Kree Cole-McLaughlin (BS), Ajith Mascarenas (Ph.D.), Vijay Natarajan (Ph.D.), Sinesio Pesco (Postdoc), Jonathan Strasser (MS).

In the ViSUS project (see the featured article in the LDRD report) we develop data streaming techniques for progressive processing and visualization of large scientific datasets. Our strategy is to exploit the coupling between time-critical algorithms and progressive multi-resolution data-structures to realize an end-to-end optimized flow of data from the original source, such as remote storage or large scientific simulation, to the rendering hardware. The implementation of this approach will enable three major visualization modalities. (i) Interactive visualization on high resolution power-walls. (ii) Interactive visualization on desktop workstations of large datasets that cannot be stored locally. (iii) Immediate monitoring of remote simulations from a desktop workstation. These modalities target multiple phases in the process of generating and exploring very large simulation datasets where real-time user interaction can increase the productivity of scientists.

Download Windows Visus Installer

San diego

Type RGB
Size 204 Gb
Width 200.000
Height 365.000

2kbit1

Type Gray
Size 8 Gb
Width 2048
Height 2048
Depth 2048

Visible male

Type RGB
Size 13 Gb
Width 2048
Height 1216
Depth 1878

Nuclear

Type Float32
Size 1 Gb
Width 600
Height 600
Depth 100
Num fields 7

Atlanta

Type RGB
Size 292 Gb
Width 320.001
Height 327.001

Phoenix

Type RGB
Size 1086 Gb
Width 720.000
Height 540.000

Hamilton

Type RGB
Size 155 Gb
Width 240.000
Height 232.000

San Francisco

Type RGB
Size 207 Gb
Width 225.000
Height 330.000

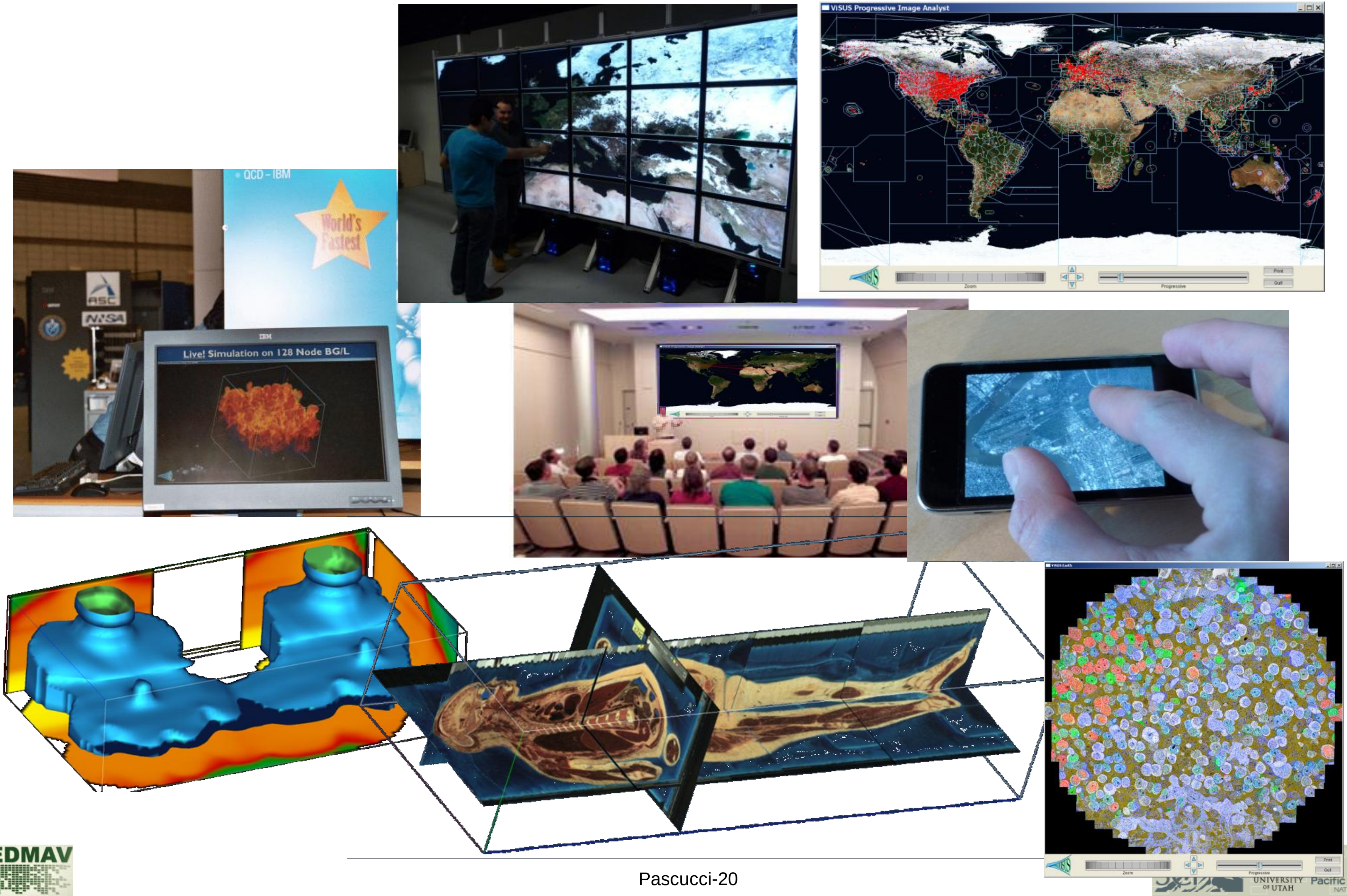
Chattanooga

Microscopy

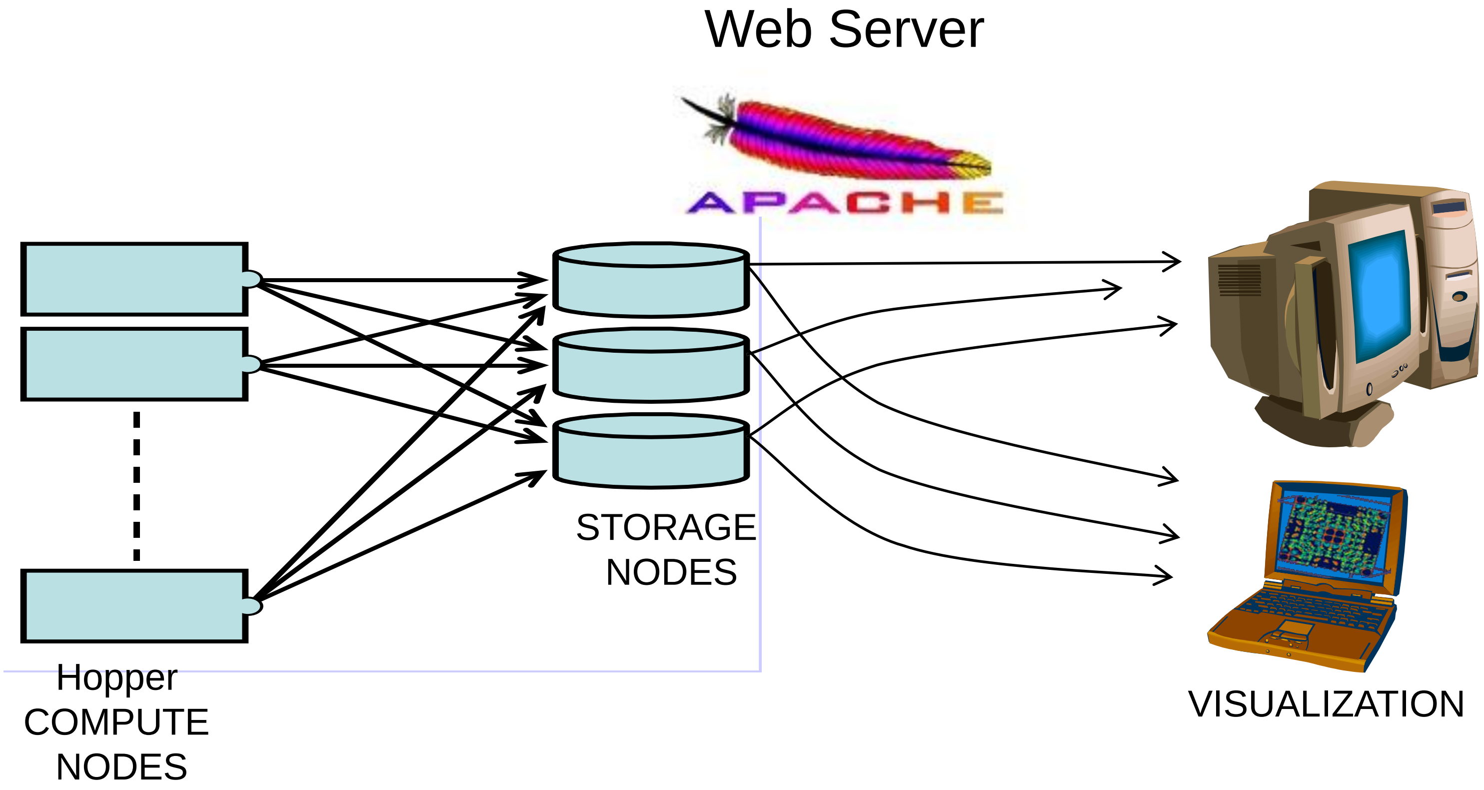
Geospatial Data Rendering on iPad

Both client and SERVER run of handheld devices, e.g. multiple iPhones can be clients and servers for each other to share information on the field

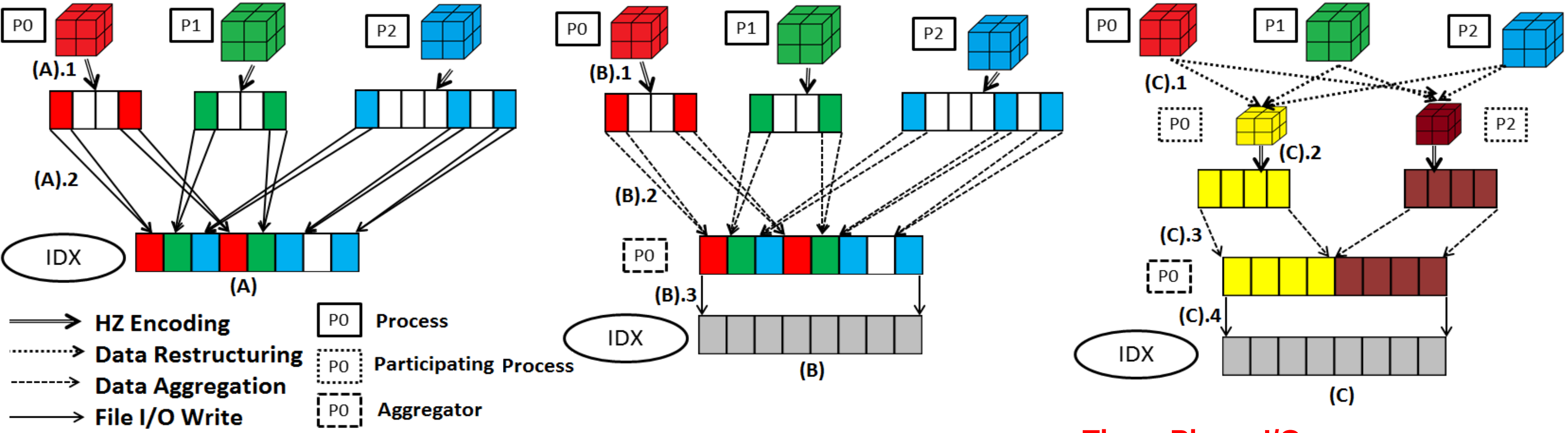
We Demonstrated Performance and Scalability in a Variety of Applications



Streaming IDX directly from large scale (S3D) simulations



The ViSUS Parallel I/O Infrastructure (PIDX) Adopts a 3–Phase Data Transfer Model



One-Phase I/O:

(A).1 HZ encoding of irregular data set leads to sparse data buffers interleaved across processes.

(A).2 I/O writes to underlying IDX file by each process, leading to a large number of small accesses to each file.

Two-Phase I/O:

(B).1 HZ encoding of irregular data set leads to sparse data buffers interleaved across processes.

(B).2 Data transfer from in-memory HZ ordered data to an aggregation buffer involving large number of small sized data packets.

(B).3 Large sized aligned I/O writes from aggregation buffer to the IDX file.

Three-Phase I/O:

(C).1 Data restructuring among processes transforms irregular data blocks at processes P0, P1 and P2 to regular data blocks at processes P0 and P2.

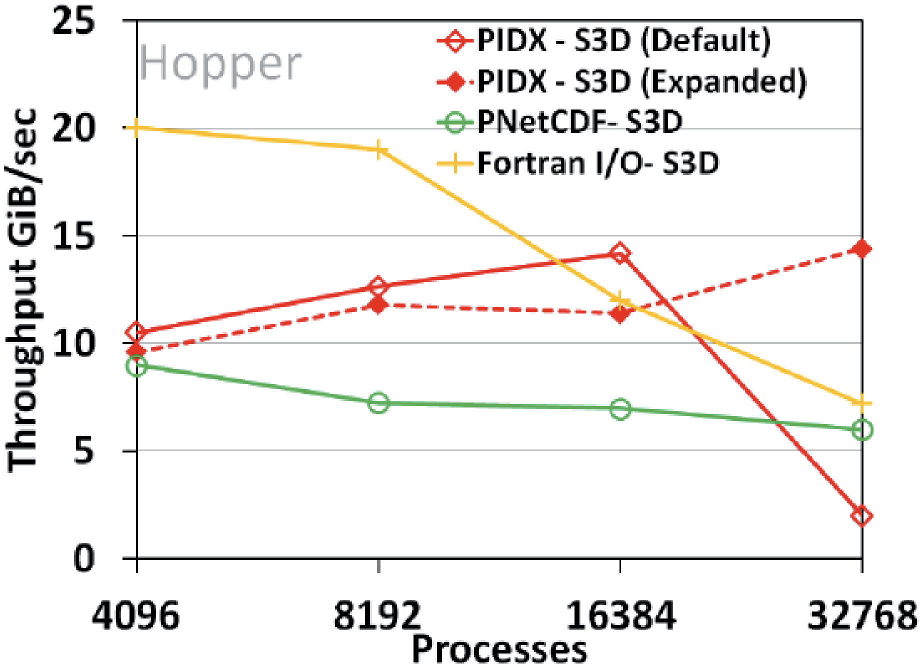
(C).2 HZ encoding of regular blocks leading to dense and non-overlapping data buffer.

(C).3 Data transfer from in-memory HZ ordered data to an aggregation buffer involving fewer large sized data packets.

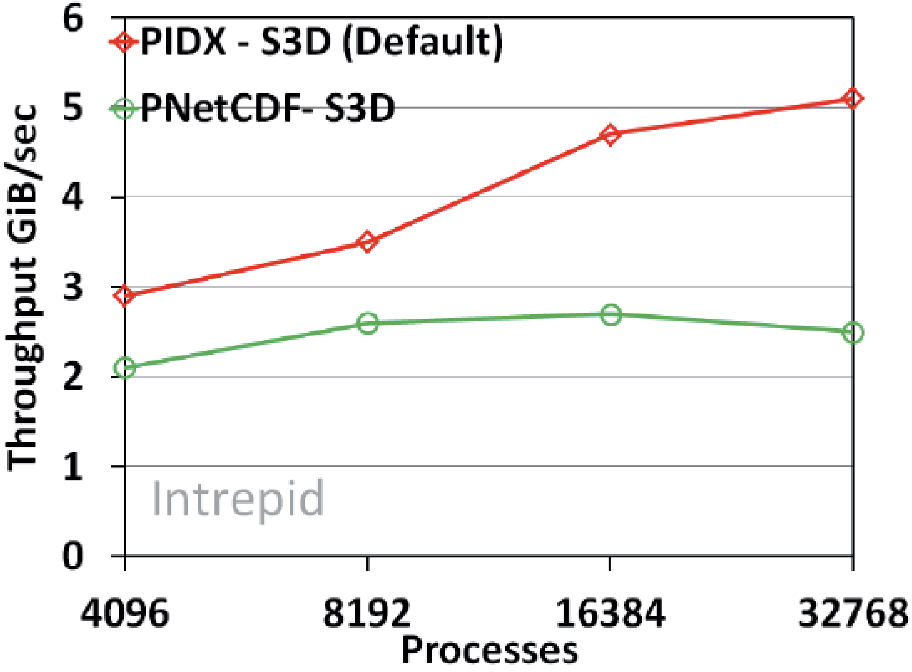
(C).4 I/O writes from aggregation buffer to a IDX file.

Strong Scaling Results Comparing PIDX Performance with PNetCDF and Fortrain I/O on Two Major Platforms

- **The PIDX Infrastructures Achieves Better Scalability than Competing Frameworks While Maintaining Advantageous Hierarchical Data Representation**

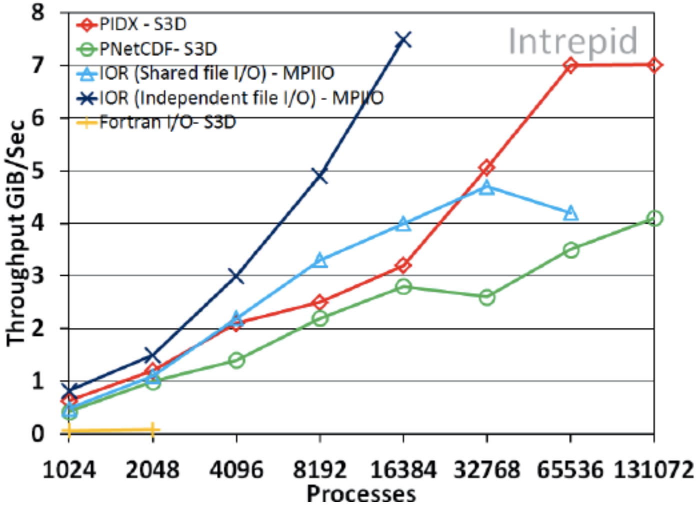


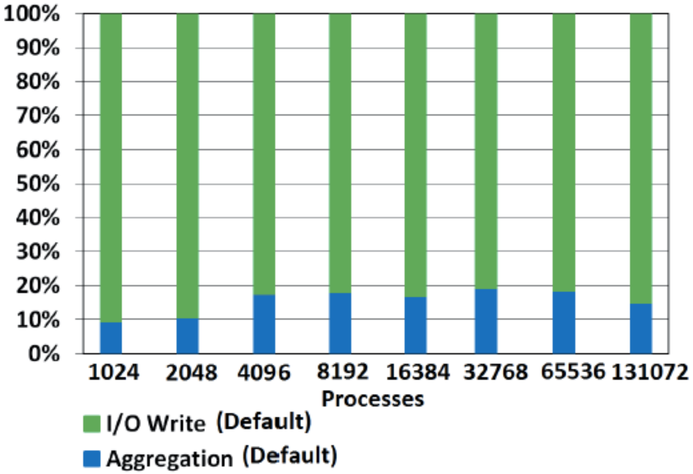
Scaling Results on Hopper
Cray XE6 architecture at NERSC (LBNL)



Scaling Results on Intrepid
BGP architecture at ALCF (ANL)

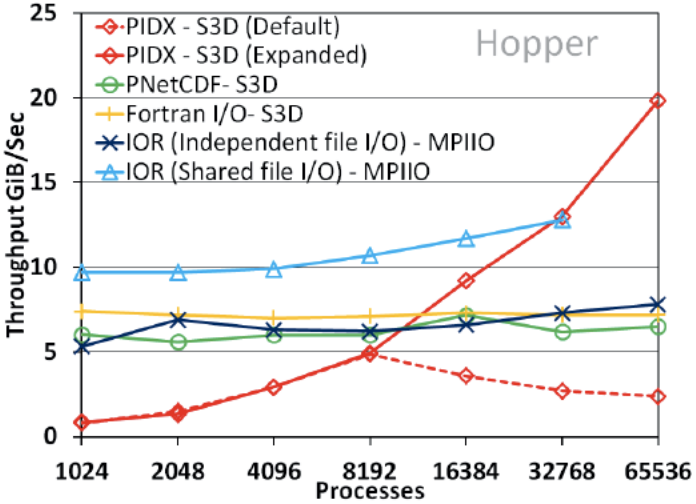
Weak Scaling Results Comparing PIDX Performance with Major Competing Techniques

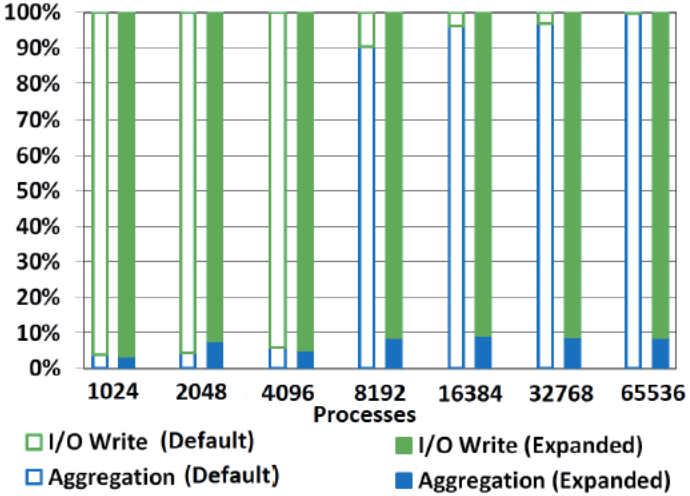




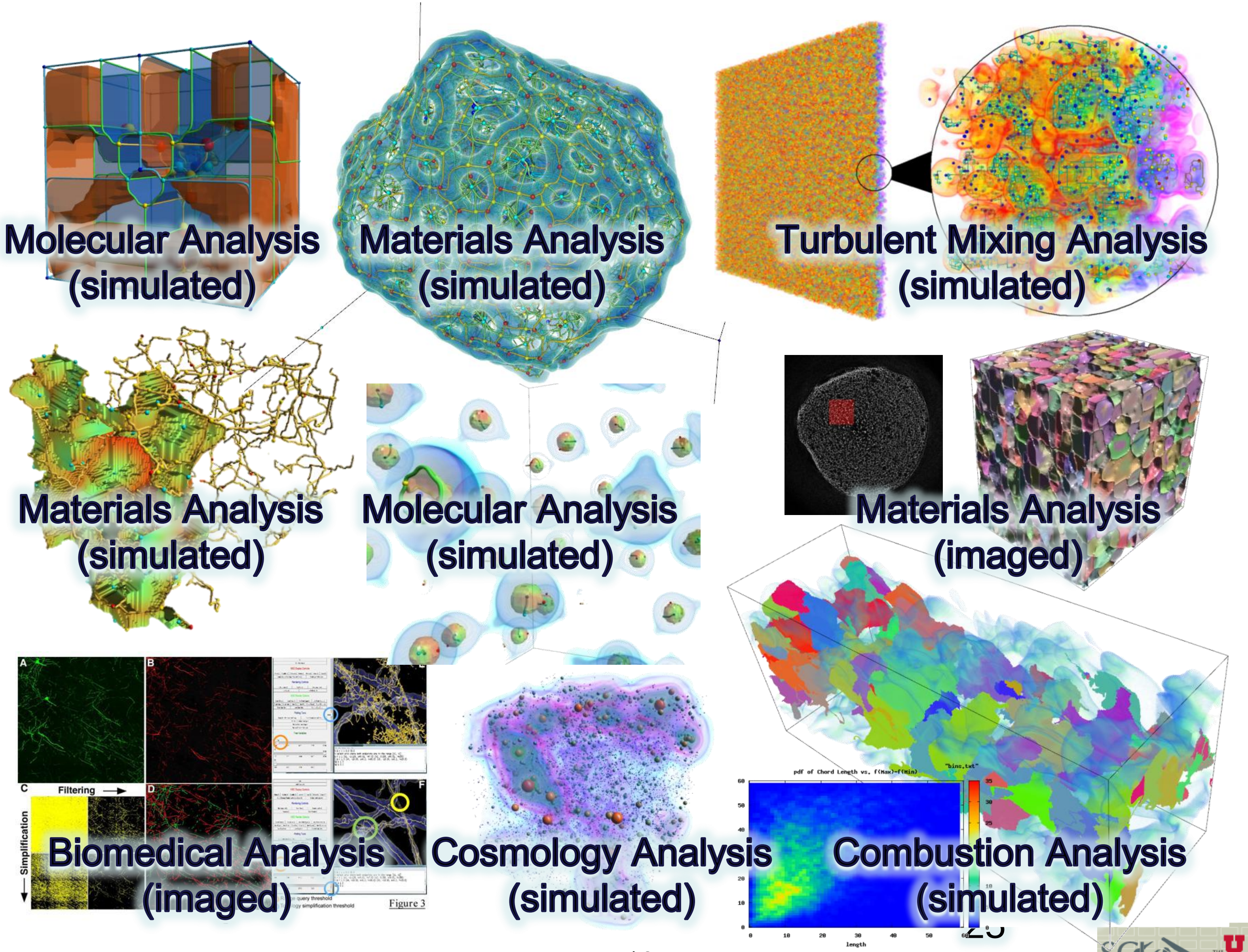
Weak Scaling Results on Intrepid BGP architecture at ALCF (ANL)

Weak Scaling Results on Hopper Cray XE6 architecture at NERSC (LBNL)





Topological Methods Have Been Successful for Analysis and Visualization of Massive Scientific Data



Molecular Analysis (simulated)

Materials Analysis (simulated)

Turbulent Mixing Analysis (simulated)

Materials Analysis (simulated)

Molecular Analysis (simulated)

Materials Analysis (imaged)

Biomedical Analysis (imaged)

Cosmology Analysis (simulated)

Combustion Analysis (simulated)