BERKELEY LAB

Lawrence Berkeley National Laboratory

U.S. DEPARTMENT OF ENERGY

UNIVERSITY OF CALIFORNIA

Lawrence Berkeley National Laboratory

Big Data Analysis and Visualization: What to LINACs and Tropical Storms Have in Common?

E. Wes Bethel (ewbethel@lbl.gov)

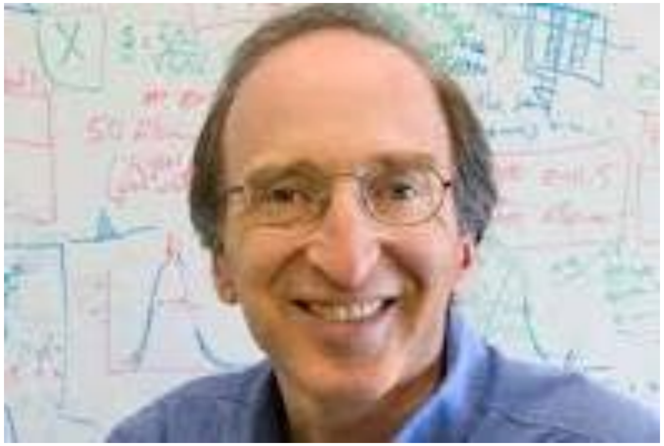
Lawrence Berkeley National Laboratory

24 August 2012

11th International Computational Accelerator Physics Conference
Rostock-Warnemünde, Germany

What's a Computer Scientist Doing at ICAP?

- A) Provide entertainment for physicists.
- B) Satisfy "diversity requirements."
- C) Saul Perlmutter declined invitation.
- D) Report on topics of potential interest.



Dr. Saul Perlmutter (LBNL)
2011 Nobel Prize (Physics)

What's a Computer Scientist Doing at ICAP?

- A) Provide entertainment for physicists.
- B) Satisfy "diversity requirements."
- C) Saul Perlmutter declined invitation.
- D) Report on topics of potential interest.

Computer & Computational Science and ICAP

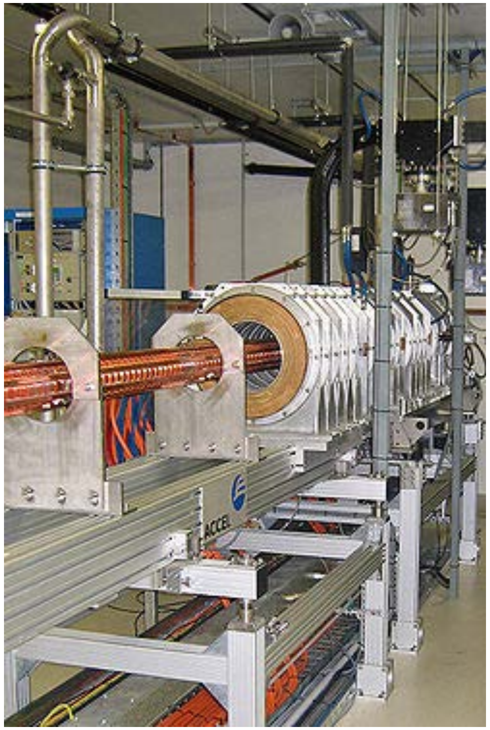
SciDAC
Scientific Discovery through Advanced Computing

- DOE programs/facilities: accelerators, supercomputing centers (and others).
- DOE SciDAC program:
  - Scientific Discovery through Advanced Computing
  - Objective: enable use of large computational platforms by computational science projects.
  - Projects: solvers, data, optimization, performance, …
- DOE-sponsored programs in data, analysis, and visualization.
  - Long history of cooperation between accelerator modeling and computer science research.

Tropical Storms and Linacs…Huh?

- LINAC = linear particle accelerator.
- Tropical storm = big, windy, wet, noisy, and messy.

- What on earth do these things have in common?



LINACs and Tropical Storms

- Climate modeling: uses simulations to model how the atmosphere, ocean, etc. respond to and behave under varying input conditions.
- LINACs: accelerator modelers use simulations to study how particle beams behave under varying beamline conditions.
- Both:
  - Produce a ton of data
  - Have lots of science questions to be answered.

Common Themes

- Data model: what variables, mesh structure, and metadata are stored.
  – Data format: how they're laid out on disk.
- Parallel I/O
- Specific science questions
  – Often identification, tracking, analysis of "features"
- Roadblocks, barriers due to combination of
  – Legacy tools don't scale
  – Don't, or can't, solve the problem

Case Studies

- Climate science
  - Feature identification, tracking, analysis in large, multivariate data
  - Analysis software infrastructure
- Laser-plasma accelerator, LINAC
  - Feature identification, tracking, analysis
  - High performance parallel I/O
  - High performance index/query

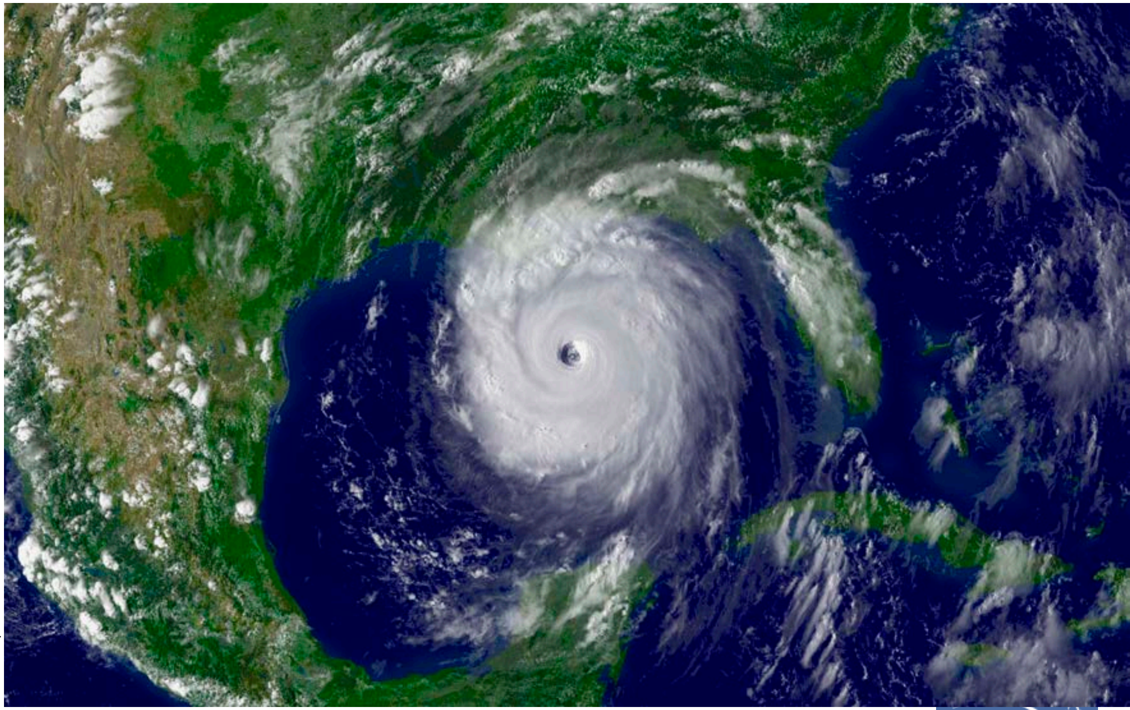
Climate Science Case Study

- Problem: want to determine how a changing climate might impact weather by looking at change in frequency and severity of extreme weather events.
- Approach: scalable analysis infrastructure that supports user-pluggable/developed "analysis kernels."

Extreme weather in a changing climate

- Severe storms
  - Hurricanes (Tropical Cyclones)
  - Extra Tropical Cyclones
  - Atmospheric Rivers
  - Mesoscale Convective Systems

- Blocking events
  - Heat waves and droughts
  - Cold snaps

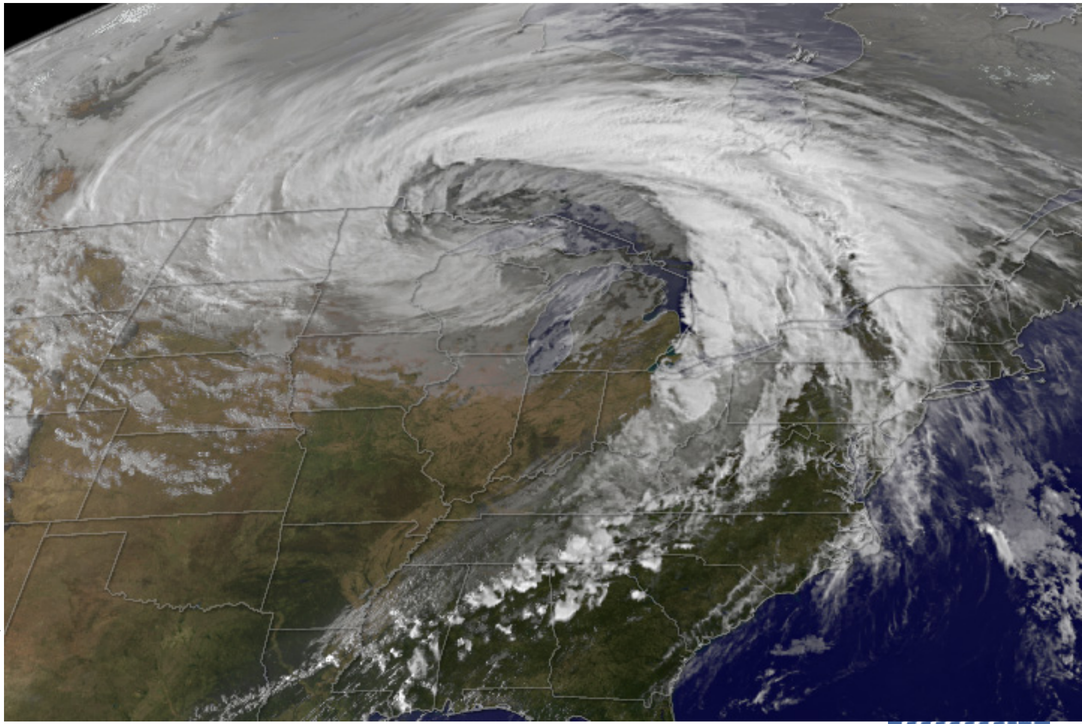


NASA/NOAA

Extreme weather in a changing climate

- Severe storms
  - Hurricanes (Tropical Cyclones)
  - Extra Tropical Cyclones
  - Atmospheric Rivers
  - Mesoscale Convective Systems

- Blocking events
  - Heat waves and droughts
  - Cold snaps

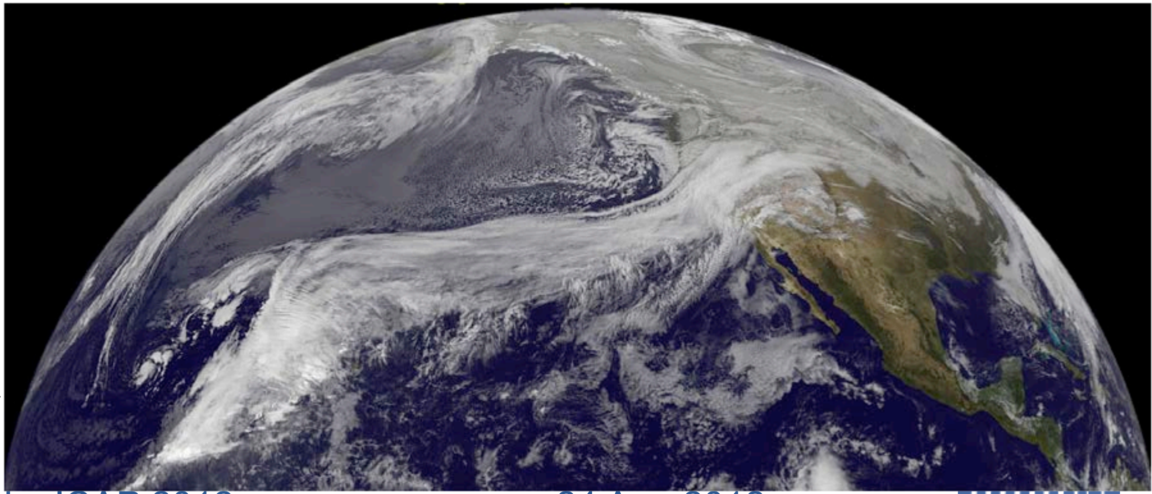


NASA GOES

Extreme weather in a changing climate

- Severe storms
  - Hurricanes (Tropical Cyclones)
  - Extra Tropical Cyclones
  - Atmospheric Rivers
  - Mesoscale Convective Systems

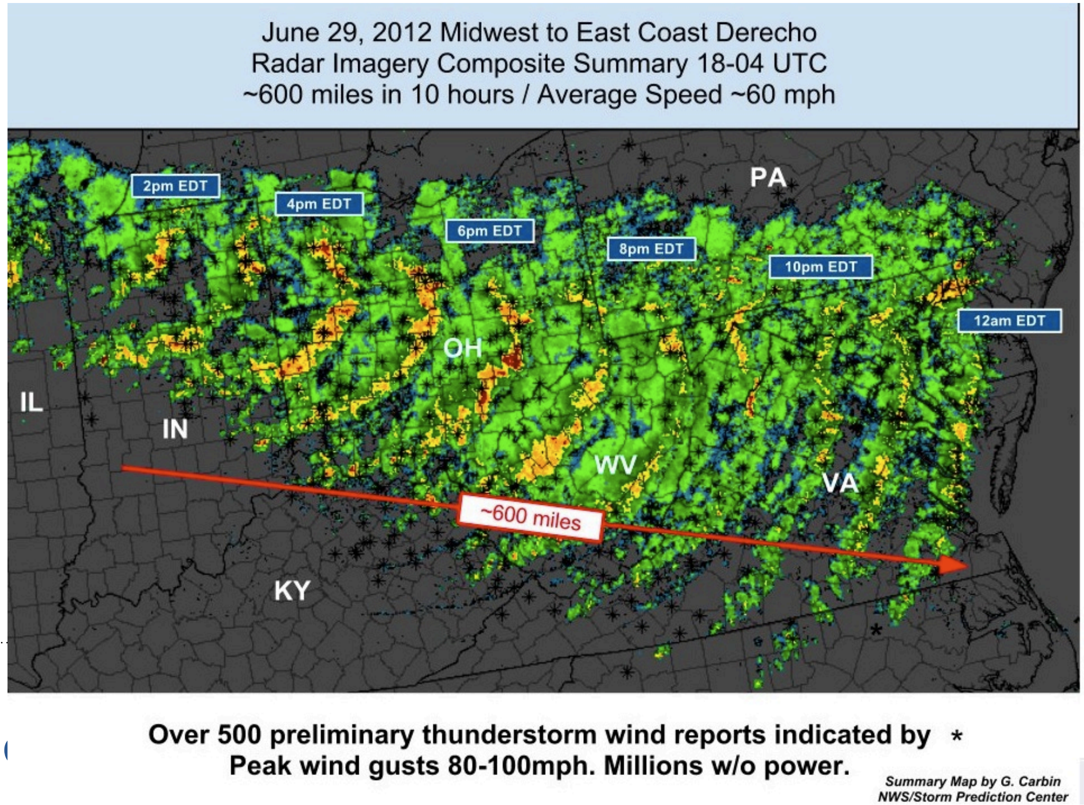
- Blocking events
  - Heat waves and droughts
  - Cold snaps



NASA

Extreme weather in a changing climate

- Severe storms
  - Hurricanes (Tropical Cyclones)
  - Extra Tropical Cyclones
  - Atmospheric Rivers
  - Mesoscale Convective Systems
- Blocking events
  - Heat waves and droughts
  - Cold snaps

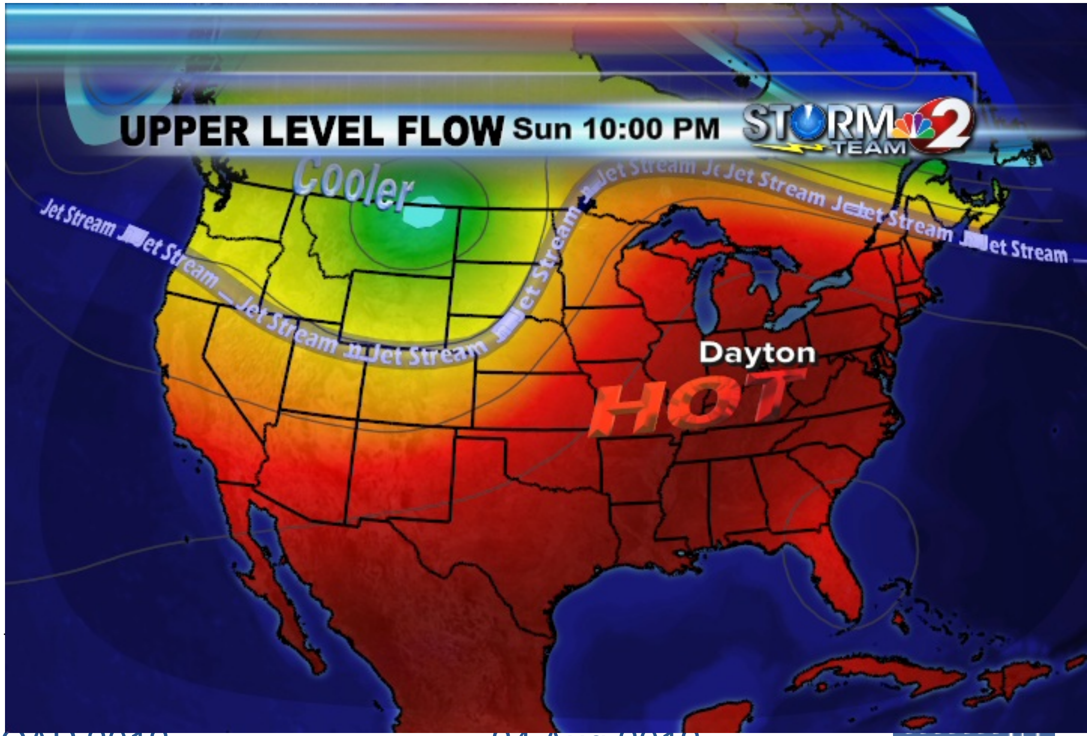


NOAA NWS

Extreme weather in a changing climate

- Severe storms
  - Hurricanes (Tropical Cyclones)
  - Extra Tropical Cyclones
  - Atmospheric Rivers
  - Mesoscale Convective Systems

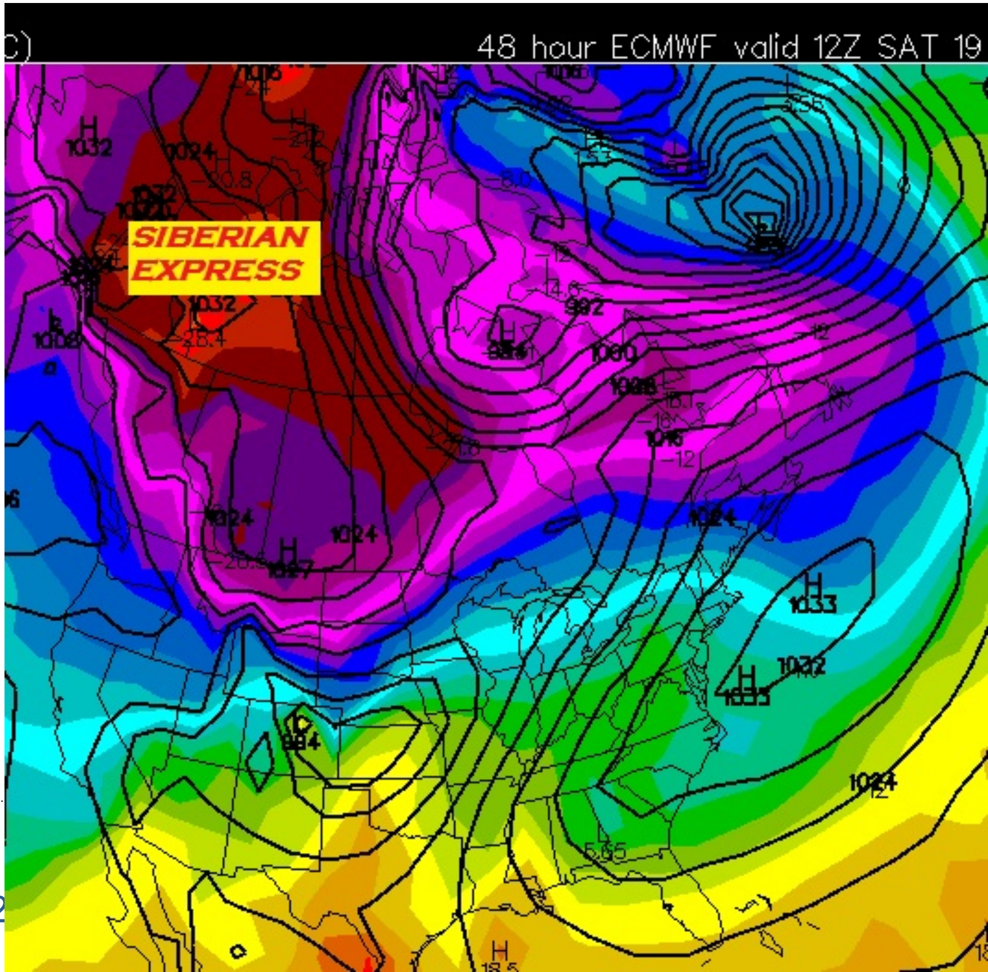
- Blocking events
  - Heat waves and droughts
  - Cold snaps



WDTN
Dayton, Ohio

Extreme weather in a changing climate

- Severe storms
  - Hurricanes (Tropical Cyclones)
  - Extra Tropical Cyclones
  - Atmospheric Rivers
  - Mesoscale Convective Systems
- Blocking events
  - Heat waves and droughts
  - Cold snaps



ECMWF
Weather underground

Challenges to indentify extreme events

- State of the art high resolution climate models can inform us about extreme weather changes, but
  - Unprecedented volumes of data need be generated.
    - We generated 100TB of output in a 26 year integration of a ~25km global atmospheric model (NCAR CAM5.1).
- Tracking extreme weather events is data intensive.
  - Scales poorly with resolution. In some cases as n^4.
    - Often high frequency (3 or 6 hourly)
  - Can be I/O bound on the input side.
  - Parallel processor tools are essential.
    - But not widely available to the climate model analyst community.

n=number of points on a horizontal direction

TECA: A Toolkit for Extreme Climate Analysis

The abstraction:

- Identifying extreme weather events in high frequency climate model output involves two steps:
    1. Search through the data for candidate events at each individual time step that meet some defined criteria.
    2. Stitch together candidate events at multiple time steps, rejecting candidates that fail continuity criteria.
- Step 1 can be very computationally intensive. But is embarrassingly parallel across time steps
- Step 2 is relatively inexpensive.

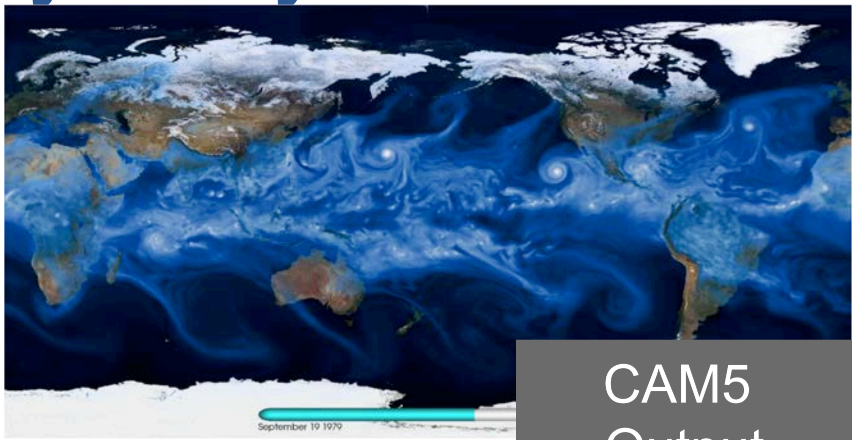
TECA Design and Execution Model

- Data parallel partitioning, execution.
- Data scatter
- Data processing
  - Each PE executes its code on a spatially disjoint block of data
  - Stores results (feature location, metadata)
- Data gather
  - Individual results gathered to one PE for final processing and analysis.

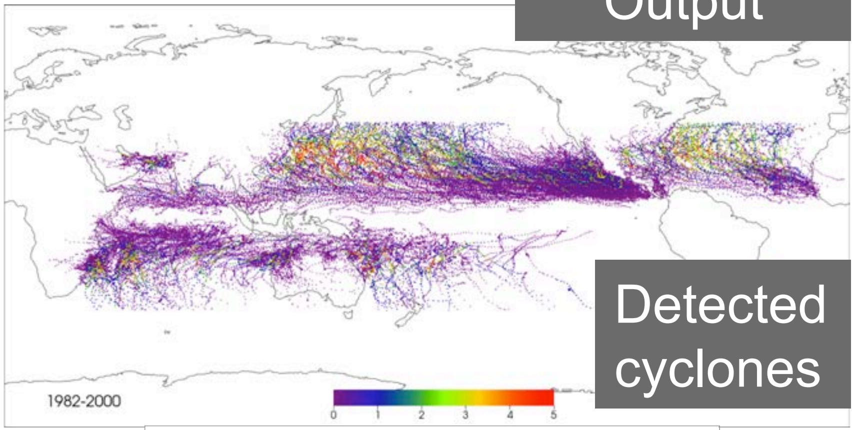
Feature Detection and Analysis: Cyclone Detection

- Science objective: quantify hurricane/cyclone characteristics in a changing climate.
- Recent CAM5 0.25° runs for 1982-2000 result in 100TB of model output.
- GFLD tracking code parallelized over time.
- Analysis time:
  - 2hrs wallclock on 7K CPUs.
  - Est. serial time: 583 days.

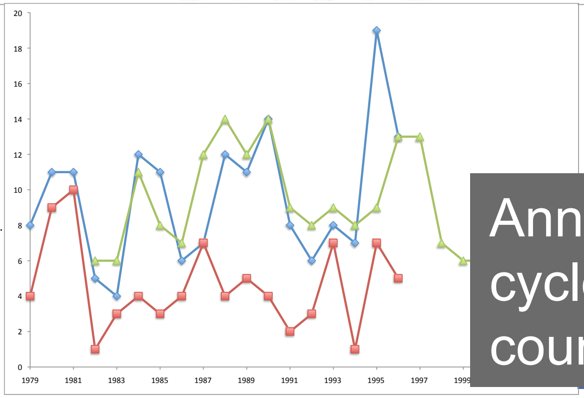
Images courtesy: M. Wehner, K. Wu, Prabhat (LBNL)



CAM5 Output



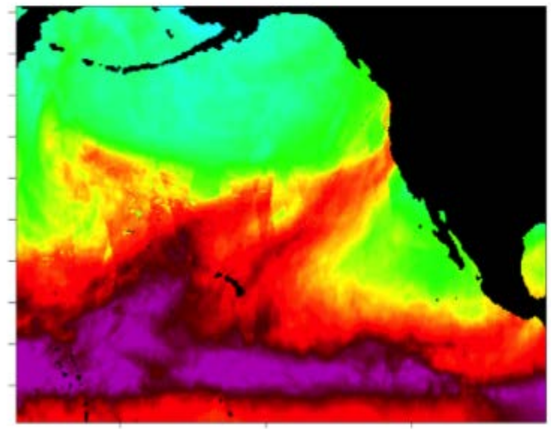
Detected cyclones

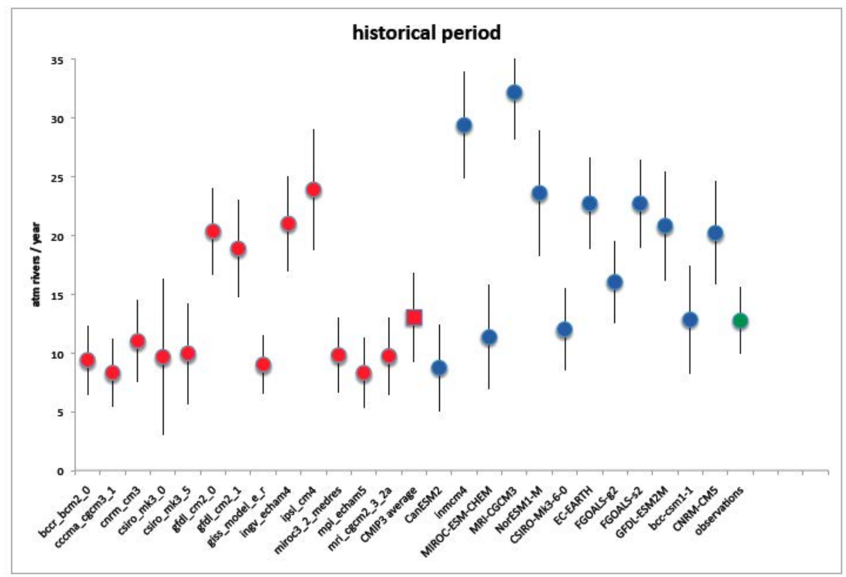


Annual cyclone counts

Atmospheric Rivers

- Science objective: quantify AR characteristics in a changing climate
- Diagnostics on AR characteristics (landfall point, length, width) added to detection procedure
- Analysis time:
  - 42 minutes to process 156 years of daily output from a single CMIP5 model using 3650 cores
  - ~0.5 second to post-process the output file to generate diagnostics and yearly statistics
  - Est. serial time: ~106 days





Climate Science Study – Main Messages

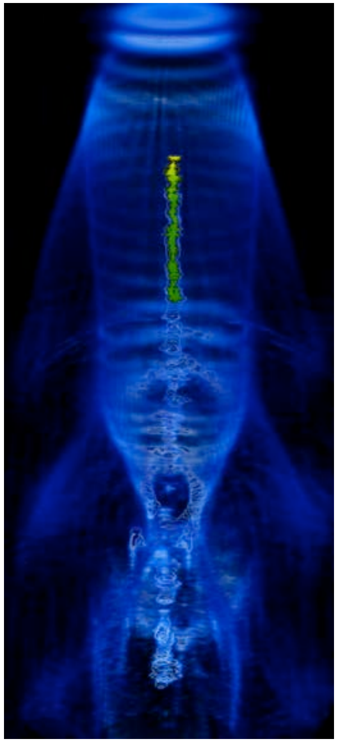
- Study of extreme weather events requires finding, tracking, analyzing "features" in data.
- Traditional approaches: infeasible.
- Data model/format issues:
  - "Community standards" in place for many applications
  - Not so much so for some applications (e.g., GCRM)
- New work: TECA
  - Scalable analysis infrastructure supports user-pluggable analysis kernels, uses "Map-Reduce"-like design and execution model.
  - Applied to several different types of extreme weather problems.
  - Open question: would this be useful to the accelerator modeling community? (e.g., particles vs. fields/meshes)

Accelerator Modeling Applications

- Laser-plasma accelerator
  - Quickly finding, analyzing, visualizing particles undergoing LPA acceleration.
- Linear accelerator
  - Overcoming I/O barrier to enable study of halo particles.
- Another PIC-based application:
  - VPIC plasma modeling, 2 trillion particle runs, I/O, analysis.

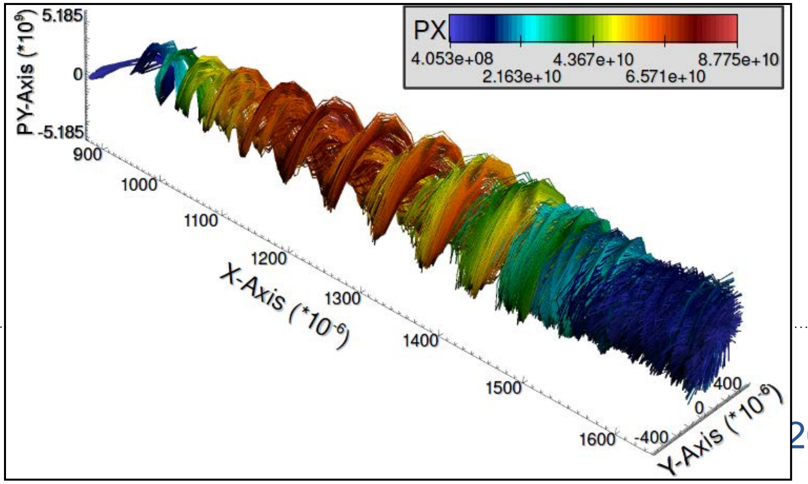
LPA Workflow

- Run simulation, dump time-varying data.
- Pick a late timestep, find particles having momentum *px > 8.872e+10.*
- Find those particles in all other timesteps.
- Connect them (particle paths), study interplay between particles and plasma, EM field.



LPA Workflow: Legacy Tool – MATLAB script

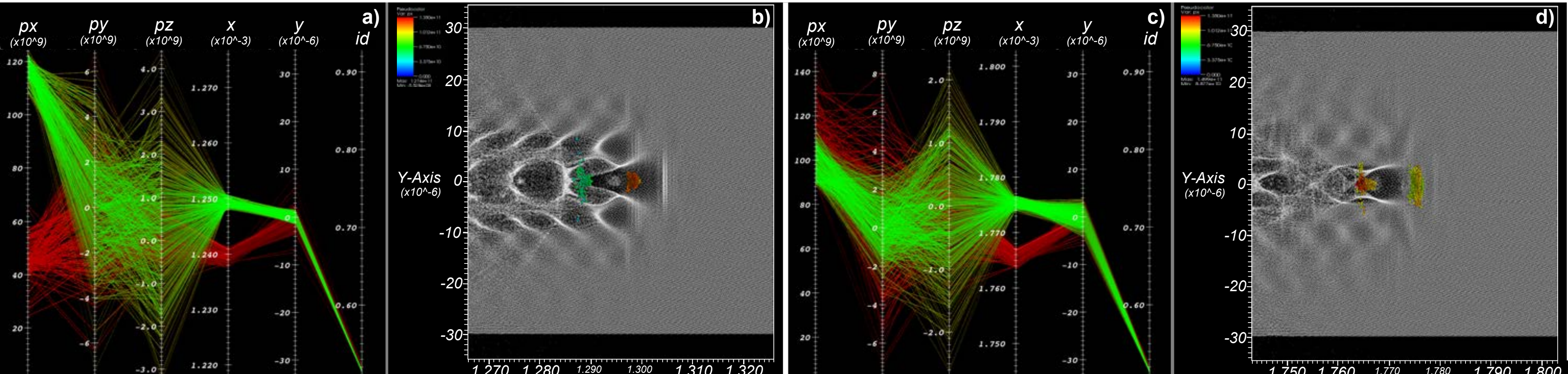
- Serial code memory footprint limit
  - Limits # of particles per timestep that can be processed.
  - ca. 2008: 4GB/core available, 5GB needed for single timestep (approx 90M particles).
- Sequential scan to look for matching particles.



HPC Visual Data Exploration, Analysis

- Multivariate range selection (query formulation)
- Parallel index, query
- Coupling to visualization application (VisIt)
- Old: hours, new: seconds.

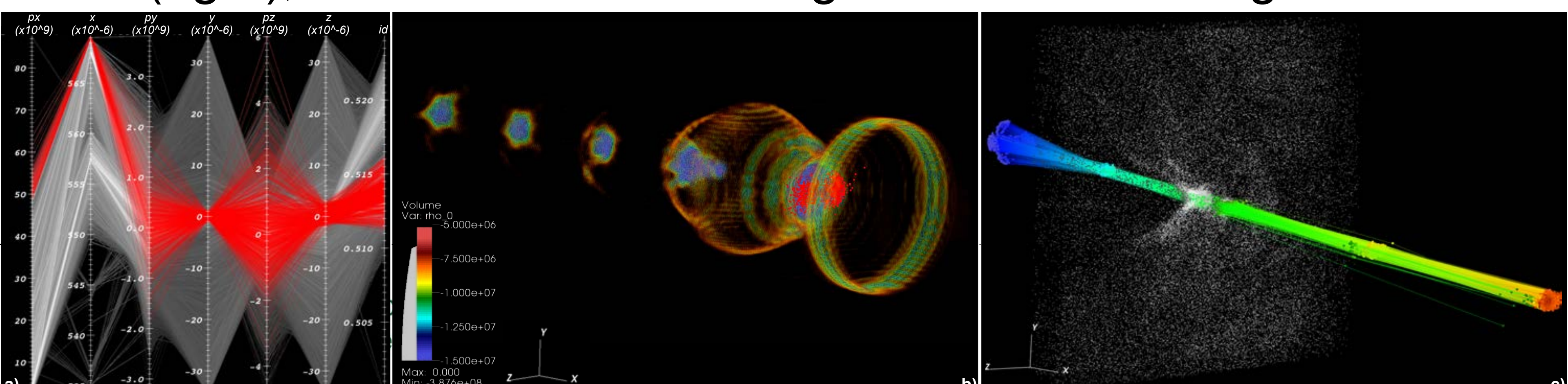
2D example: t=27 (left), t=37 (right). Two bunches selected (red, green): both have high velocity and spatial coherence.



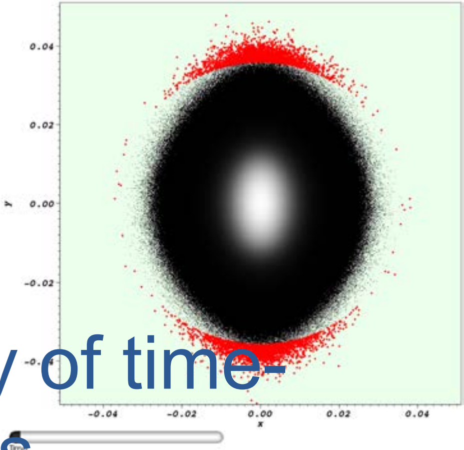
HPC Visual Data Exploration, Analysis

More information: Rübel et al., 2008. High Performance Multivariate Visual Data Exploration for Extremely Large Data. Supercomputing 2008.

3D example. High velocity bunch selected (left), shown in 3D context (middle). All particles of that bunch traced over time (right), color shows increasing acceleration along trace.



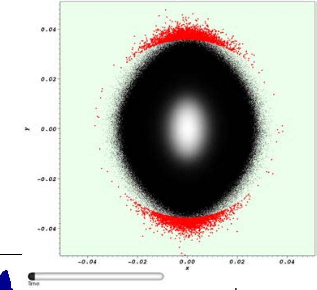
LINAC Modeling

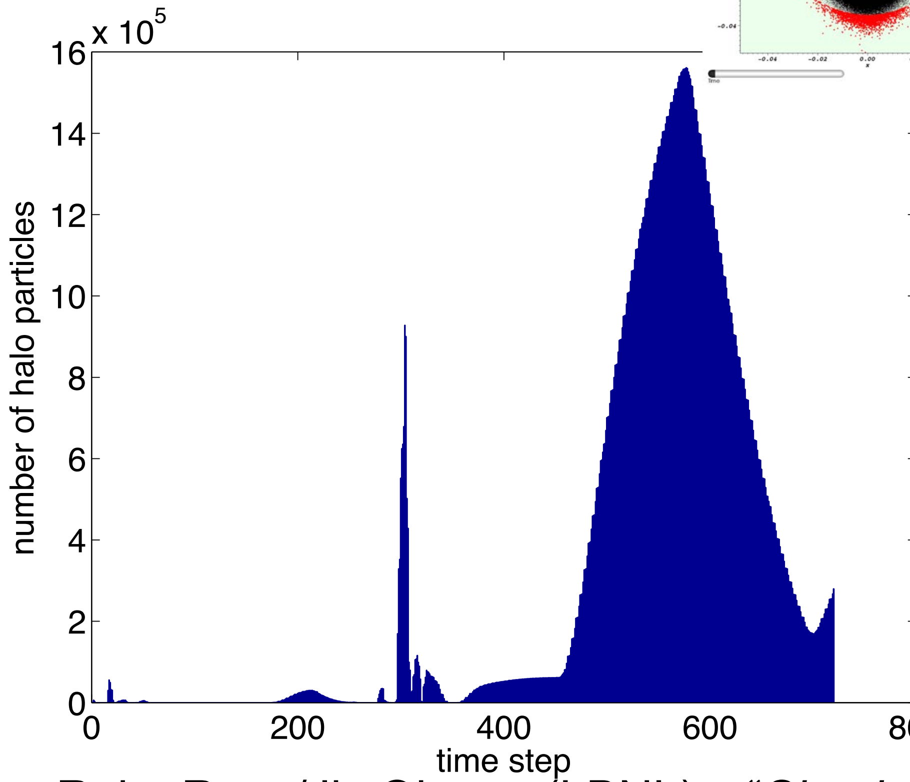


- Problem: I/O bottlenecks prevent study of time-evolving phenomena, like halo particles, particularly in high-resolution runs.
- Approach:
  - Add advanced I/O capability to IMPACT-Z,
  - Apply advanced index/query and query-driven visual data analysis to study halo particle formation, evolution.
- Impact: first-ever 1B particle runs show suspected, but never before seen, halo particles in late stages of simulation.

LINAC Modeling, ctd.

- Enabled code to generate 50TB of data on 10,000 franklin cores at 5GB/s
  - Previously, this code not capable of writing that much data.
- Queried 50TB dataset in ~10 seconds on 3,000 hopper cores.
- Provided quantitative feedback to accelerator designers.
  - Number of halo particles per timestep.
  - Suspected, but never before seen.

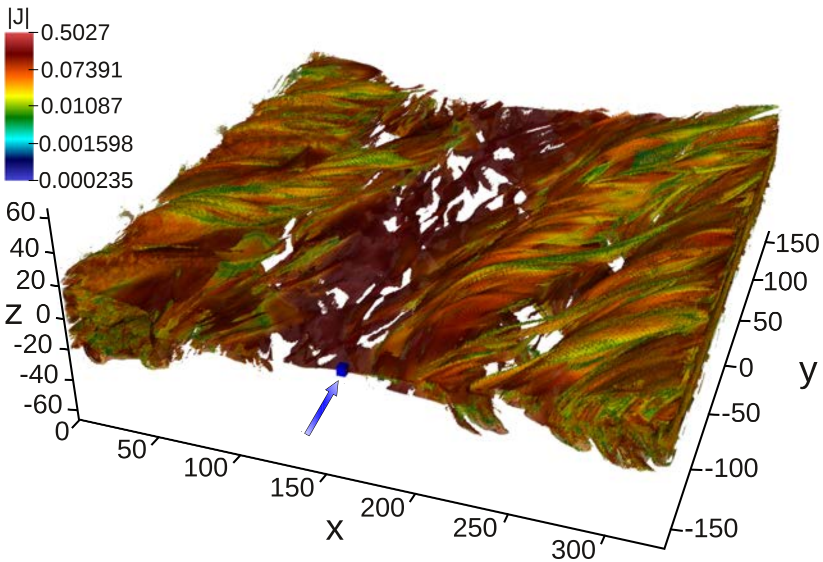


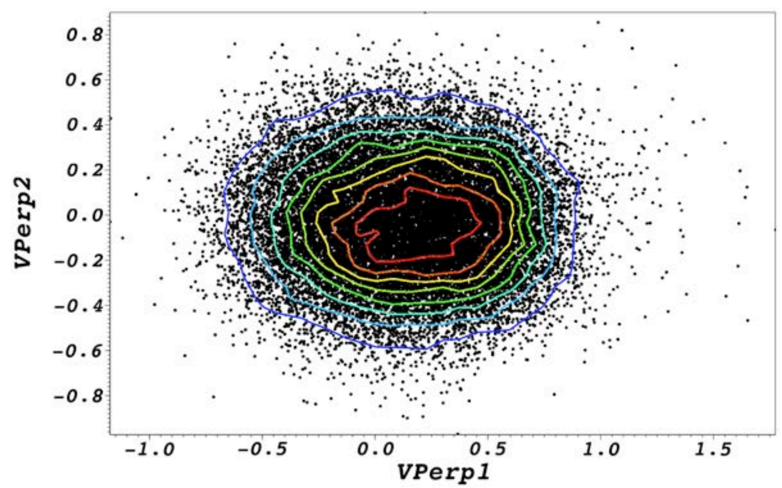


Rob Ryne/Ji Qiang (LBNL): "*Simulated electron bunch seems to scatter more in the latter half of the beamline. This was expected but we didn't confirm this quantitatively before this collaboration*"

Parallel I/O and Analysis of a Trillion Particles

- Trillion particle plasma physics simulation conducted on 120,000 cores @NERSC
- Enhanced Parallel HDF5 obtained peak 35GB/s, and 80% sustained I/O rate
- FastBit was used to index 30TB timestep in 10 minutes and query in 3 seconds
- Software enabled scientists to examine and gain insights from the trillion particle dataset for the first time:
    - Confinement of energetic particles by the flux ropes
    - Asymmetric distribution of particles near the reconnection hot-spot





Magnetic reconnection from a plasma physics simulation (Left). Scientists were able to query and find an asymmetric distribution of particles near the reconnection event (Right) using our software tools.

Evolution of Data Model, Format

- Circa 2008/2009:
  - VORPAL dumps post-processed to produce separate index files
- Circa 2011:
  - IMPACT-Z instrumented to do parallel I/O
  - Indexing performed at write
  - Indices and data payload inside HDF5 file
- Circa 2012:
  - VPIC instrumented to do parallel, collective I/O
  - Indexing performed at write
  - Hybrid-parallel capable query

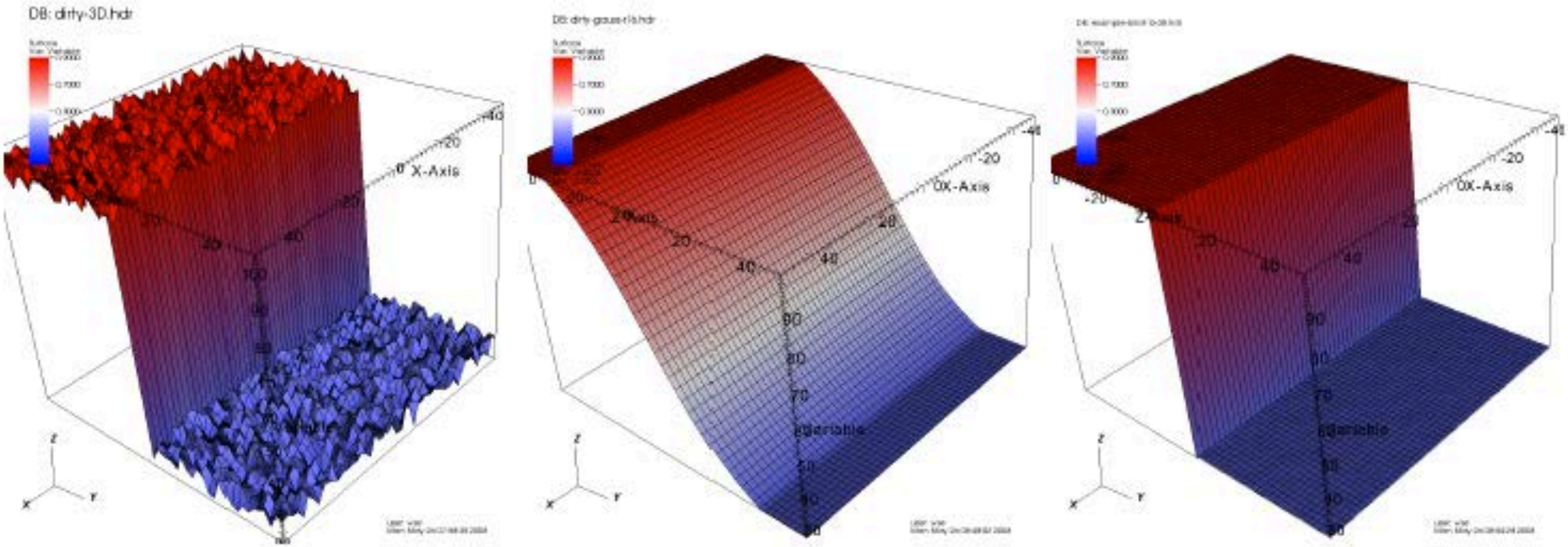
LPA/LINAC Common Themes

- Data models, formats, and parallel I/O
  - – End-to-end view of requirements, design
    - What are analysis requirements?
    - Best way to write data may not be best for reading.
  - – Doing 100K-way parallel I/O is not trivial
- Want: HPC I/O, analysis, to be “trivial” and “transparent”.
  - – Making progress, but not there yet.

Transparency Issue: Complex Platforms and Optimal Performance

- With increasing complexity of computational platforms:
  - There are more “knobs” to tune
  - Increasingly difficult to achieve optimal performance
  - “Auto-tuning” provides one avenue to help
- Want this capability to be “transparent” to applications.

Example: The Bilateral Filter



(a) Synthetic, noisy dataset. (b) Gaussian smoothing ($r = 16$) (c) Bilateral smoothing ($r = 16$, $\sigma = 15\%$)

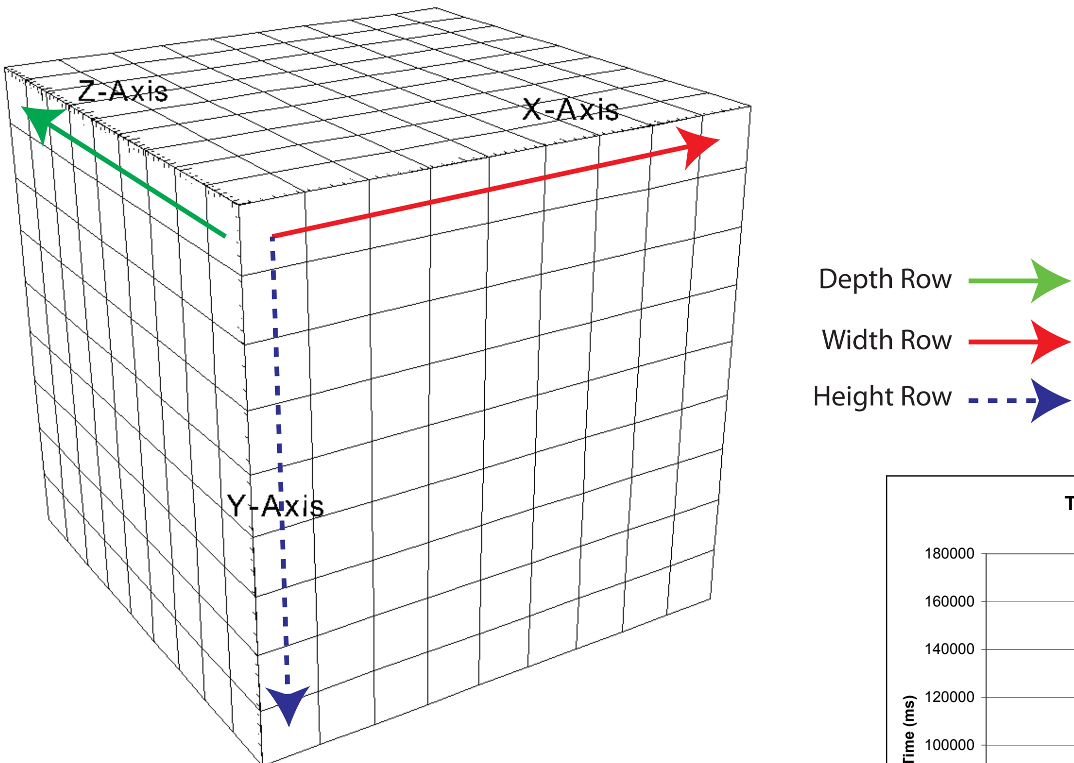
$$d(i) = \frac{1}{k(i)} \sum g(i, \bar{i}) c(i, \bar{i})$$

g(i) – spatial weight
c(i) – photometric weight

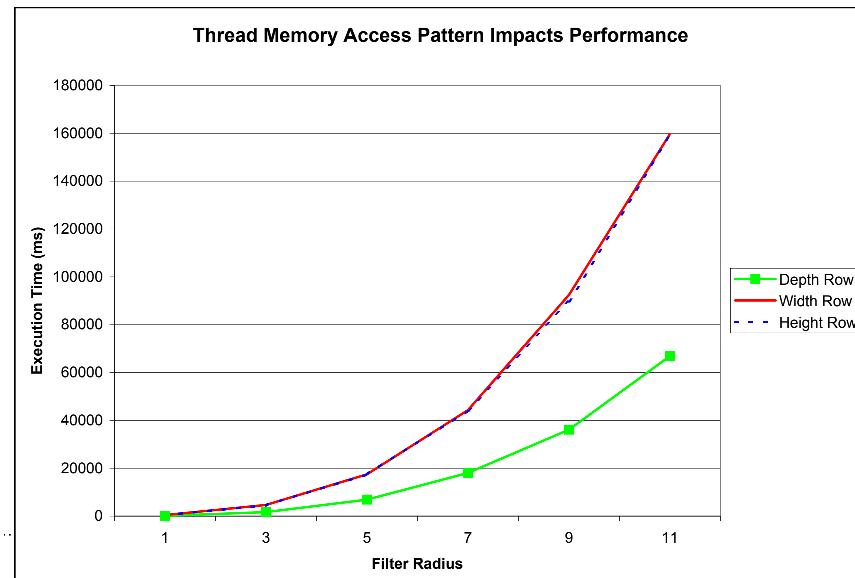
GPU Stencil Code Optimization/Tuning

- Algorithm: 3d bilateral filter (stencil code)
- Platform: GPU
- Questions:
  - Does CUDA thread block size/shape influence performance?
  - Can use of device-specific capabilities improve performance?
  - Does inner-loop processing order influence performance?

Does Thread Processing Order Matter?

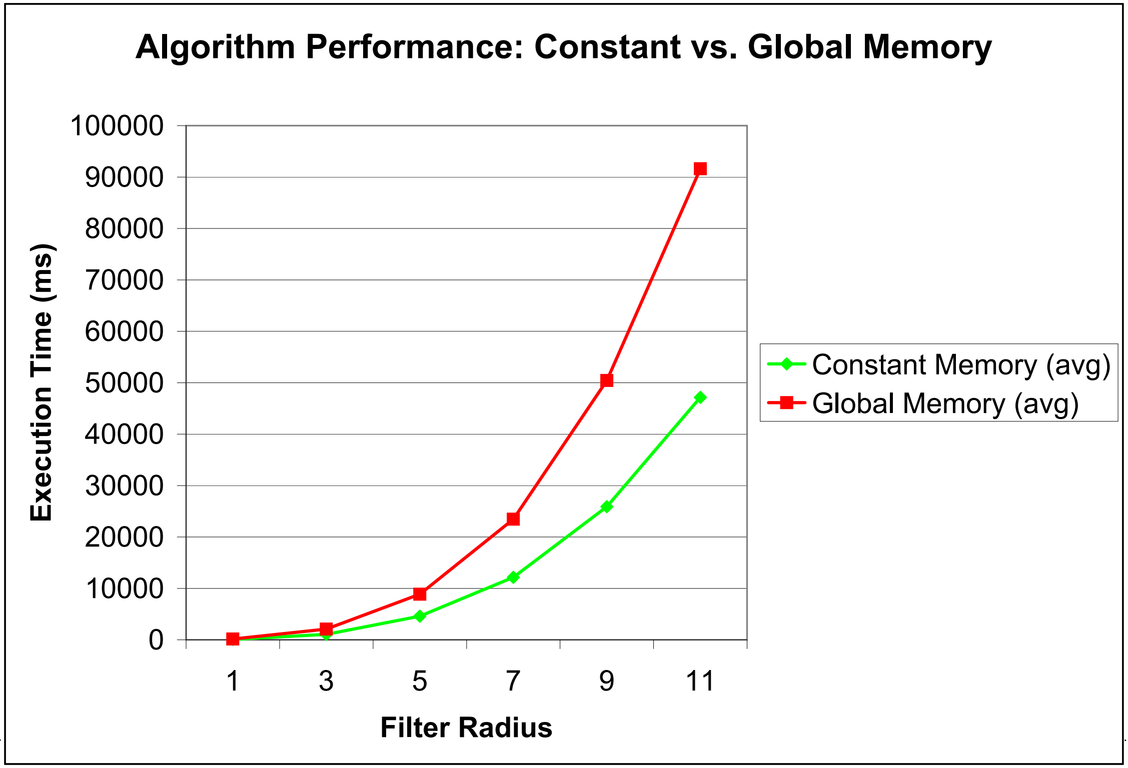


Yes! 2x difference.



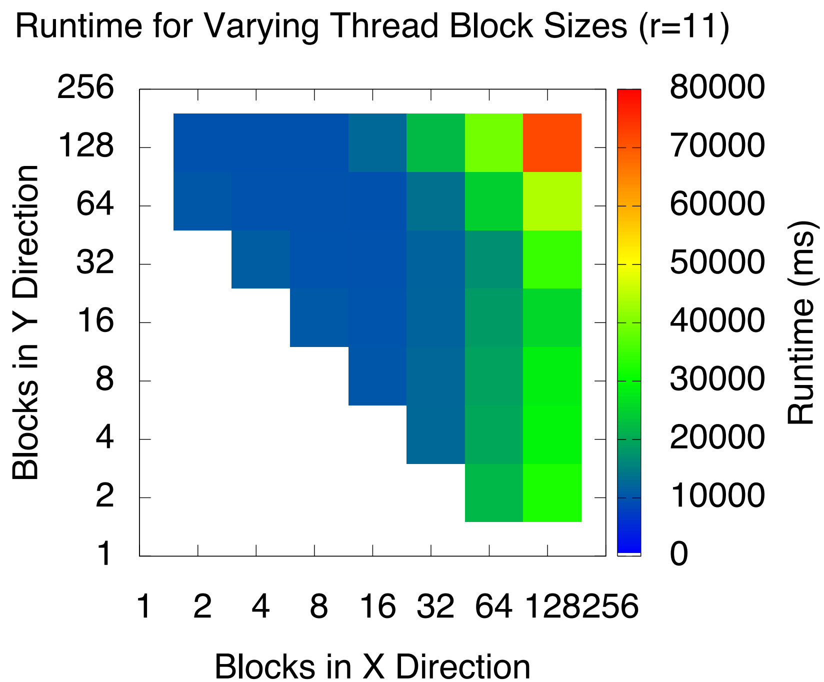
Does a Device-specific Feature Help?

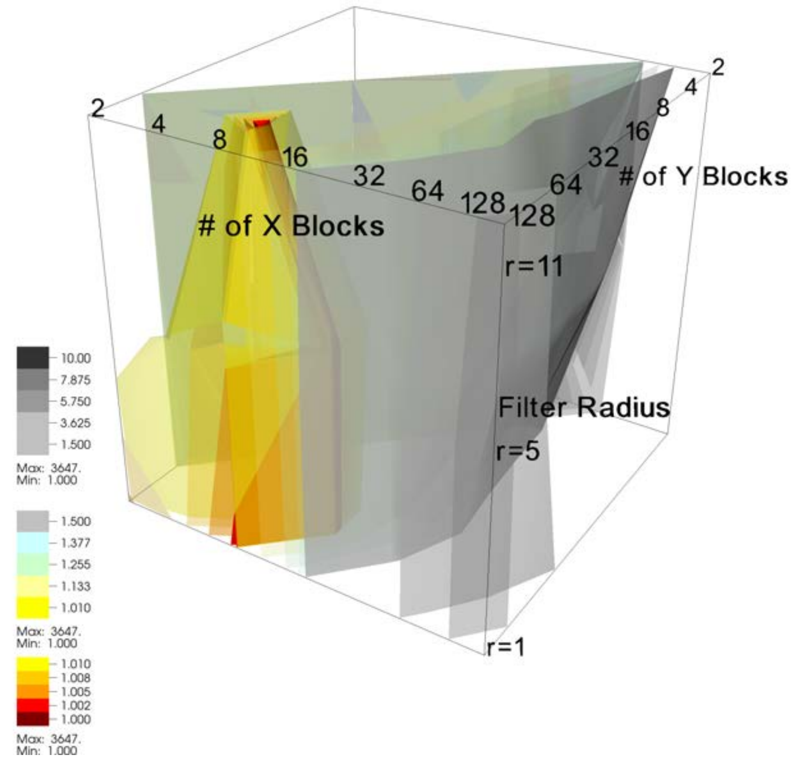
- Storing filter weights in “constant” rather than “global” memory.



Yes! Another 2x. But what about other GPUs?

Does Thread Block Size/Shape Matter?





Runtime at r=11 (left), at r=[1,5,11] (right).
~7.5x performance variation.

Study Results

- Bottom line:
  - 30x performance gain through a combination of tuning, algorithmic design options, and use of device-specific features.
- CUDA Block size: 7.5x
- Device-specific capabilities: 2x
- Processing order: 2x

Many-/multi-core Parallel Volume Rendering Optimization and Tuning – E. Wes Bethel (LBNL)

ASCR- Visualization Research Highlight

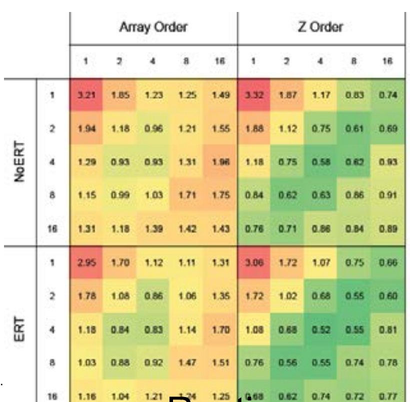
Objectives

- Apply principles of "auto-tuning" to a staple visualization algorithm, raycasting volume rendering.
- Gain better understanding of tunable algorithmic parameters and their impact on performance on multi-/many-core platforms.

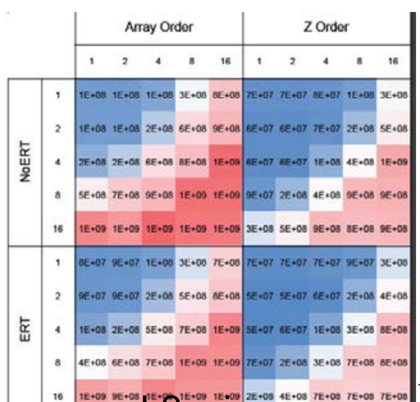
Impact

- The settings that produce the best performance vary from problem to problem, platform to platform, often in a non-obvious way.
- Informs design/implementation of petascale- and exascale-class visualization/analysis codes.

On the GPU, the fastest runtimes (left, green blocks) occur with medium-sized work units. However, and surprisingly, small-sized work blocks give the best memory utilization (right, blue blocks) but the worst absolute runtime. Conclusion: "best runtime" is not always correlated to "best use of memory."



Runtime



L2 misses

Progress & Accomplishments (FY11)

- **First-ever of its type: an extensive, comprehensive study on multi-core CPUs and many-core GPUs to measure performance impact (runtime, memory utilization) of tunable algorithmic parameters. We observe up to 250% variation in runtime on m-core CPUs, and 470% variation on m-core GPUs.**
- **Used results from an early version of this study for 216,000-way hybrid-parallel runs on JaguarPF for optimal performance.**
- **Publication: International Journal of High Performance Computing Applications (in press 2012)**

Other Ongoing Work – *In Situ* Processing

- Don't store full res data:
  - Instead, perform analysis/vis while data still in simulation memory.
  - Avoids increasingly expensive I/O.
- Several flavors (two axes):
  - (API) Custom, general, hybrid.
  - (execution model) Concurrent, co-processing
- Example: Warp3D and *in situ*
  - Kernels in Fortran, python "exterior"
  - Version 1 *in situ* – use OpenDX
  - Version 2 *in situ* – use VisIt

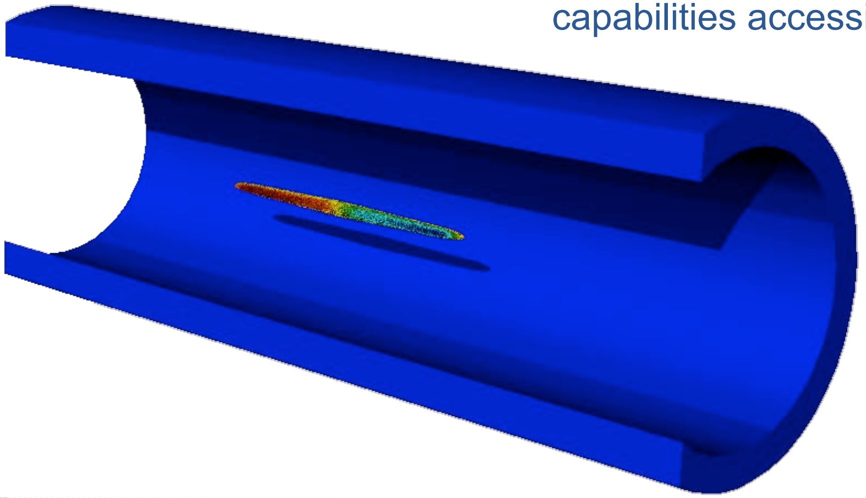
In Situ: Warp3D and VisIt

Science Problem

- Warp3D is an advanced simulation code developed by the Heavy-Ion Fusion Science Program
- The current *in situ* visualization approach has several limitations:
  - Legacy system OpenDX is no longer supported
  - Visualization is performed by a single compute core
  - The data often needs to be reduced for the visualization

Approach/Impact

- Couple general purpose *in situ* technology, VisIt, to accelerator modeling code, Warp3D.
- Using VisIt for *in situ* data analysis and visualization:
  - Enables *in situ* processing of the complete data in parallel
  - Avoids reduction and communication of the data for the purpose of visualization
  - Makes new advanced analysis capabilities accessible to Warp3D



Summary and Conclusions

- Increasing computational horsepower provides opportunities for better simulations
  - More data, more complex data
- Existing approaches, legacy tools, often incapable of meeting needs

Summary and Conclusions, ctd.

- Meeting “big data challenge” requires considering:
  - End-to-end requirements, not just initial store
  - Data model/format issue is central
  - Often requires taking into account increasingly complex computational platforms.
- Examples here show progress towards meeting big data challenges.

More Information

- Berkeley Lab Visualization Group
  - http://vis.lbl.gov/
- SciDAC Scalable Data Management, Analysis, and Visualization Center
  - http://www.scidac-sdav.org/
- ExaHDF5 Project
  - Prabhat (LBNL), prabhat@lbl.gov


CAUTION
YOU HAVE REACHED THE
LAST PAGE
OF THE INTERNET
TURN OFF YOUR BROWSER AND GO BACK TO WORK
THERE'S NOTHING ELSE TO SEE HERE