Big Data
in Scientific Domains

Arie Shoshani
Lawrence Berkeley National Laboratory

BES Workshop
August 2012
The Scalable Data-management, Analysis, and Visualization (SDAV) Institute

2012-2017

Arie Shoshani (PI)

Co-Principal Investigators from:

<table>
<thead>
<tr>
<th>Laboratories</th>
<th>Universities</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANL</td>
<td>GTech</td>
</tr>
<tr>
<td>LBNL</td>
<td>NCSU</td>
</tr>
<tr>
<td>LLNL</td>
<td>NWU</td>
</tr>
<tr>
<td>ORNL</td>
<td>OSU</td>
</tr>
<tr>
<td>LANL</td>
<td>UCD</td>
</tr>
<tr>
<td>SNL</td>
<td>Rutgers</td>
</tr>
<tr>
<td>Kitware (Industry)</td>
<td>UUtah</td>
</tr>
</tbody>
</table>

http://sdav-scidac.org/
Outline

• Emerging challenges with Big Data in scientific domains (5 min)
• Examples of current approaches and solutions (15 min)
• Questions regarding handling of experimental data (2-3 min)
Scientific Data Management, Analysis, and Visualization

- Applications examples
  - Climate modeling
  - Combustion
  - Fusion
  - Nuclear

- Algorithms, techniques, and software
  - Representing scientific data – data models, metadata
  - Managing I/O – methods for removing I/O bottleneck
  - Accelerating efficiency of access – data structures, indexing
  - Facilitating data analysis – data manipulations for finding patterns and meaning in the data
  - Visual analytics – help understand data visually
A Typical Scientific Investigation Process

- **Current practice – data intensive tasks**
  - Runs large-scale simulations on large supercomputers
  - Dump data on parallel disk systems
  - Export some of the data to archives
  - Move data to users’ sites – usually selected subsets
  - Perform data manipulations and analysis on mid-size clusters
  - Collect experimental / observational data
  - Move experimental / observational data to analysis sites
  - Perform comparison of experimental/observational to validate simulations
  - Iterate
A typical Scientific Investigation

lots of Data Movement (GBs – TBs)

Site A
- Simulation Machine
- Parallel Storage
- Archive

Site B(i)
- Analysis Machine
- Shared storage

Site C
- Experimental/Observational data
## Exascale Systems: Potential Architecture

<table>
<thead>
<tr>
<th>Systems</th>
<th>2009</th>
<th>2018</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Peak</td>
<td>2 Pflop/sec</td>
<td>1 Eflop/sec</td>
<td>O(1000)</td>
</tr>
<tr>
<td>Power</td>
<td>6 Mwatt</td>
<td>20 Mwatt</td>
<td></td>
</tr>
<tr>
<td>System Memory</td>
<td>0.3 Pbytes</td>
<td>32-64 Pbytes</td>
<td>O(100)</td>
</tr>
<tr>
<td>Node Compute</td>
<td>125 Gflop/sec</td>
<td>1-15 Tflop/sec</td>
<td>O(10-100)</td>
</tr>
<tr>
<td>Node Memory BW</td>
<td>25 Gbytes/sec</td>
<td>2-4 Tbytes/sec</td>
<td>O(100)</td>
</tr>
<tr>
<td>Node Concurrency</td>
<td>12</td>
<td>O(1-10K)</td>
<td>O(100-1000)</td>
</tr>
<tr>
<td>Total Node Interconnect BW</td>
<td>3.5 Gbytes/sec</td>
<td>200-400 Gbytes/sec</td>
<td>O(100)</td>
</tr>
<tr>
<td>System Size (Nodes)</td>
<td>18,700</td>
<td>O(100,000-1M)</td>
<td>O(10-100)</td>
</tr>
<tr>
<td>Total Concurrency</td>
<td>225,000</td>
<td>O(1 billion)</td>
<td>O(10,000)</td>
</tr>
<tr>
<td>Storage</td>
<td>15 Pbytes</td>
<td>500-1000 Pbytes</td>
<td>O(10-100)</td>
</tr>
<tr>
<td>I/O</td>
<td>0.2 Tbytes/sec</td>
<td>60 Tbytes/sec</td>
<td>O(100)</td>
</tr>
<tr>
<td>MTTI</td>
<td>Days</td>
<td>O(1 day)</td>
<td></td>
</tr>
</tbody>
</table>

Scaling simulations generates a data volume challenge (PBs)

Site A

Simulation Machine
Parallel Storage
Archive

Site B(i)

Analysis Machine
Shared storage

Site C

Experimental/Observational data

subset
subset
subset
What Else Can be Done?

- Perform some data analysis and visualization on simulation machine (in-situ)
- Reduce Data and prepare data for further analysis (in-situ)
Bottlenecks/Problems for handling Big Data

- **Simulation Machine**
  - In-situ Analysis, Visualization
  - In-situ Data reduction, Indexing
  - Monitoring simulations progress in real-time

- **Parallel Storage**
  - I/O slows down the computation
  - No automated archiving management
  - Automated data provenance (original data)

- **Archive**
  - Selection of subsets based on content
  - Reliable and effective data movement
  - Automated data provenance (derived data)

- **Site B(i)**
  - In-situ Data reduction, Indexing
  - Selection of subsets based on content
  - Reliable and effective data movement

- **Analysis Machine**
  - Tools for real-time analysis and visualization

- **Site A**
  - In-situ Analysis, Visualization
  - Selection of subsets based on content
  - Reliable and effective data movement

- **Site C**
  - Experimental/Observational data
  - Validation of models using experimental data

- **Shared Storage**
  - Automated data provenance (original data)
  - Automated data provenance (derived data)
Some Solutions are Emerging

(1) In-situ Analysis and Visualization
(2) In-situ Analysis Data reduction and Indexing
(3) Monitoring simulations progress in real-time
(4) I/O slows down the computation
(5) Automated archival management
(6) Selection of subsets based on content
(7) Selective, reliable, and effective data movement
(8) Tools for real-time analysis and visualization
(9) Validation of models using experimental data (some)
(10) Automated data provenance (some)
Data Analysis

- **Two fundamental aspects**
  - **Pattern matching**: Perform analysis tasks for finding known or expected patterns
  - **Pattern discovery**: Iterative exploratory analysis processes of looking for unknown patterns or features in the data

- **Ideas for the exascale**
  - Perform **pattern matching** tasks in the simulation machine
    - “In situ” analysis
  - Prepare data for **pattern discovery** on the simulation machine, and perform analysis on mid-size analysis machine
    - “In-transit” data preparation
    - “Off-line” data analysis
The SDAV institute tools: Scalable Data Management, Analysis, and Visualization

**SDAV Goals:**
- to actively work with application teams to assist them in achieving breakthrough science
- to provide technical solutions in the data management, analysis, and visualization regimes that are broadly used by the computational science community running on Leadership Class Machines
- To use existing robust tools to the extent possible and develop/adapt tools on an as-needed basis

---

**Data Management tools**
- In Situ Processing and Code Coupling
  - ADIOS
  - Glean
- Indexing
  - FastBit
- In Situ Data Compression
  - ISABELLA
- Parallel I/O and File Formats
  - PnetCDF
  - BP-files
  - HDF5

**Data Analysis tools**
- Statistical and Data Mining Techniques
  - NU-Minebench
- Importance-Driven Analysis Techniques
  - Domain-Knowledge Directed
  - Geometry Based
- Topological Methods
  - In Situ Topology
  - Feature-Based Analysis
  - High-Frequency Analysis and Tracking

**Visualization tools**
- Parallel visualization
  - Visit
  - ParaView
- VTK-m framework
- Flow Visualization Methods
- Rendering
- Ensembles, Uncertainty, and Higher-Dimensional Methods
Example:
Capturing I/O functions
to control I/O efficiency
ADIOS: Adaptable I/O System

- Overview: service-oriented architecture:
  - Allows plug-ins for different I/O implementations
  - Abstracts the API from the method used for I/O
- Simple API, almost as easy as F90 write statement
- Synchronous and asynchronous transports supported with no code changes
- Change I/O method by changing XML file only
- ADIOS buffers data
- ADIOS allows multiple transport methods per group
High Performance I/O with ADIOS

Technology

- ADIOS: adaptable I/O framework
  - Provides portable, fast, scalable, easy-to-use, metadata rich output with a simple API
  - Allows plug-ins for different I/O implementations
  - Abstracts the API from the method used for I/O
  - http://www.nccs.gov/user-support/center-projects/adios

Result/Impact

- Allowed scientists to analyze their data efficiently within their allotted computing time
- Made multi-pass analysis developed in the course of the investigation feasible
- Led to new understanding of the flow phenomena that impact flame stabilization
- Fundamental science that facilitates safe, efficient design of gas turbines for hydrogen-rich fuels

"Volume rendering of square jet in cross-flow case showing Temperature, Fuel (H2), and intermediate (HO2) species mass fractions."

Image from:
Example:

Monitoring simulations progress in real-time
Monitoring simulations progress in real-time (Initially inspired a Fusion project)

Monitoring saves cycles

- **Automate the monitoring pipeline**
  - Use workflow
    - Monitoring of large-scale simulations must be dynamic
    - transfer of simulation output to remote machines
    - execution of conversion routines
    - image creation, data archiving

- **Requirements for ease-of-use**
  - Use dashboard
    - A way to view monitoring information from anywhere
    - A way to see graphs, images, and movies
    - A way to see provenance and move data to user’s site
    - A way to perform statistical analysis, compare graphs, etc.
The Kepler Workflow Engine

- Kepler is a workflow execution system based on Ptolemy (open source from UCB)
- Our work is in the development of components for scientific applications (called actors)
Real-time visualization and analysis capabilities on dashboard

visualize and compare shots
Example:

Use of parallel visualization for remote multi-discipline scientists
(virtual reactor)
Analysis and Visualization of Light Water Reactors

Application:
- CASL: Consortium for Advanced Simulation of Light Water Reactors

Simulation Goal:
- Use leadership-class computing for engineering design and analysis of next generation reactors, life extension, and higher fuel burnup (“virtual reactor”)

Requirement: Remote Interactive Analysis
- Interactive 3D visualization of disparate simulation data
- Support for large numbers of remote, multi-discipline scientists and engineers
- Multi-physics, multi-material simulations across a broad range of both spatial and temporal scales

Challenges
- Large data generated by multiple, high resolution simulation codes
  - High core count simulations
  - Multi-physics, multi-materials
  - Whole reactor core
- Size of data requires multi-site, remote access
Analysis and Visualization with VisIt

**Technology**

- **VisIt**: Turnkey, general purpose tool for large data analysis and visualization
- Client/server architecture for remote visualization on analysis clusters and supercomputers
- Demonstrated scaling to > 100K cores
- Plugin architecture for flexibility, and extensibility

**Result/Impact**

- Client-server architecture allows scientists to easily perform analysis and visualization remotely using analysis clusters, or supercomputers.
- Allowed scientists to explore simulation results at a variety of scales, including: full reactor core, materials, and structural components.
- Scripted, comparative visualizations were critical in validating new computational techniques for power distribution.

Full core, and pin assembly transport simulations rendered in VisIt using the parallel volume renderer.

Validation study comparing assembly reactor core power calculations from two different codes.
Example:

Indexing to Select Subsets Based on Content facilitates interactive visualization
Selection of subsets based on content

- Find the HEP collision events with the most distinct signature of Quark Gluon Plasma
- Find the ignition kernels in a combustion simulation
- Track a layer of exploding supernova

These are not typical database searches:

- Large high-dimensional data sets (1000 time steps X 1000 X 1000 X 1000 cells X 100 variables) – each time step can have 100 billion data values
- No modification of individual records during queries, i.e., append-only data
- M-Dim queries: 500 < Temp < 1000 && CH3 > 10^{-4} && …
- Large answers (hit thousands or millions of records)
- Seek collective features such as regions of interest, histograms, etc.

- Other application domains:
  - real-time analysis of network intrusion attacks
  - fast tracking of combustion flame fronts over time
  - accelerating molecular docking in biology applications
  - query-driven visualization
FastBit: accelerating analysis of very large datasets

- Most data analysis algorithm cannot handle a whole dataset
  - Therefore, most data analysis tasks are performed on a subset of the data
  - Need: very fast indexing for real-time analysis

- FastBit is an extremely efficient compressed bitmap indexing technology
  - Indexes and stores each column separately
  - Uses a compute-friendly compression techniques (patent 2006)
  - Improves search speed by 10x – 100x than best known bitmap indexing methods
  - Excels for high-dimensional data
  - Can search billion data values in seconds

- **Size:** FastBit indexes are modest in size compared to well-known database indexes
  - On average about 1/3 of data volume compared to 3-4 times in common indexes (e.g. B-trees)
Flame Front Tracking with FastBit

Flame front identification can be specified as a query, efficiently executed for multiple timesteps with FastBit.

Cell identification
Identify all cells that satisfy user specified conditions:
“600 < Temperature < 700 AND HO₂concentr. > 10⁻⁷”

Region growing
Connect neighboring cells into regions

Region tracking
Track the evolution of the features through time
Query-Driven Visualization

- Collaboration between data management and visualization technologies
  - Use FastBit indexes to efficiently select the most interesting data for visualization
- **Above example: laser wakefield accelerator simulation**
  - VORPAL produces 2D and 3D simulations of particles in laser wakefield
  - Finding and tracking particles with large momentum is key to design the accelerator
  - Brute-force algorithm is quadratic (taking 5 minutes on 0.5 mil particles), FastBit time is linear in the number of results (takes 0.3 s, 1000 X speedup)
Example:

Client – Server Remote Visualization
Visualization of Laser Back Scatter

**Application:**
- Laser-induced fusion at LLNL
- Laser back scatter modeling by Dr. Steve Langer

**Simulation Goal: Understand Laser Back Scatter**
- Two laser beams impacting a deuterium and tritium target
- Determine amount and direction of back scattered energy

**Requirement: Visualizing the time dependent behavior**
- Visualizing input energy and back scatter energy over time
  - Understand how the back scatter is formed
  - Understand the orientation and intensity of the back scattered energy

**Challenges**
- Extreme data size generated by high resolution simulation (220 billion cells)
- Correlating multiple, complex, 3d phenomenon over time

A burst of back scatter energy traveling from left to right
Side-by-Side Volume Visualizations of Time Dependent Behavior

**Technology**

- Ability to view data using a client/server architecture using VisIT
- Ability to quickly generate side-by-side animations of key physics quantities to understand time dependent behavior
- Automated movie generation makes it easy for the user to see time evolution of data with different views and transfer functions.

**Result/Impact**

- Client/server architecture allows the user to view his data on his desktop without moving the data
- Ability to interactively set transfer function to bring out features of interest at key points in the simulation
- This is the first time scientists could see how the backscatter was forming. Orthogonal slices was inadequate, since this was a 3d phenomenon. This helps them design the target and laser pulse timing.

Reduced input beam corresponds to high back scatter

![Input beam and Back scatter graphs](image-url)
Example:

In Situ Data Reduction
Promising Ideas for larger scale data: 
and User-Assisted Data Reduction

In situ analysis incorporates analysis routines into the simulation code. This technique allows analysis routines to operate on data while it is still in memory, potentially significantly reducing the I/O demands.

One way to take advantage of in situ techniques is to perform initial analysis for the purposes of data reduction. With help from the application scientist to identify features of interest, we can compress data of less interest to the scientist, reducing I/O demands during simulation and further analysis steps.

The feature of interest in this case is the mixture fraction with an iso value of 0.2 (white surface). Colored regions are a volume rendering of the HO2 variable (data courtesy J. Chen (SNL)).

By compressing data more aggressively the further it is from this surface, we can attain a compression ratio of 20-30x while still retaining full fidelity in the vicinity of the surface.

Contact: Kwan-Liu Ma
Example:

parallelization of large-scale 3D movies
Analysis and Visualization of Magnetic Reconnection

**Application:**
- Simulation of magnetic reconnection by Bill Daughton (LANL) and Homa Karimabadi (UCSD)

**Simulation Goal:**
- To understand the 3D evolution of tearing modes – a type of plasma instability that spontaneously produces magnetic reconnection while giving rise to topological changes in the magnetic field.

**Requirement: Remote Interactive Analysis**
- Interactive 3D visualization of simulation data
  - Particle data
  - Mesh data
- Comparison with theoretical expectations
- Rapid exploration due to limited availability of supercomputer to run large simulations

**Challenges**
- Large data size generated by high resolution simulation
  - Simulation on 98304 cores
  - 6.4 billion cells
  - 1.5 trillion particles
  - 57 TB data
- Only remote access to the supercomputer
- Lack of dedicated visualization resources
Remote Visualization with ParaView

Technology

- **ParaView**: general purpose data analysis and visualization tool focused on large data
- Client/server architecture for remote visualization on supercomputers
- Designed to be extensible: I/O routines and domain-specific algorithms developed by science teams

Result/Impact

- Allowed scientists to remotely analyze and visualize their data when it is not possible to copy locally
- Allowed scientists “to rapidly explore the grid data to understand the 3D evolution of magnetic reconnection”
- As expected, a spectrum of tearing instabilities develops which interact, forming new current sheets and triggering secondary tearing instabilities

Isosurface of particle density colored by current density
Summary

• Many of the tools in the SDAV institute have been developed over many years and are very robust and well-documented

• Some of the tools have been designed to take advantage of high level parallelism

• Such tools have been used for multiple scientific domains, but often require collaboration between application scientists and tool experts; that is the role of SDAV

• For anticipated future needs tools are being enhance in several ways:
  • Scale tools for high parallelization levels
  • Adapt tools to take advantage of new hybrid hardware (CPUs + GPUs)
  • Minimize data movement between nodes
  • Adapt tools for in situ processing and analysis to provide early insight of the generated data
  • Compress and index data in situ for both in situ and post-processing analysis
What Else Can be Done?

- Perform some data analysis and visualization on simulation machine (in-situ)
- Reduce Data and prepare data for further analysis
What about experimental data?

- Move all data to a data center that keep tracks and archives data?
What about experimental data?

- Or perform local analysis, data reduction, and visualization (in-situ)
- Then move data to a shared facility for management and distribution of reduced experimental data
THE END