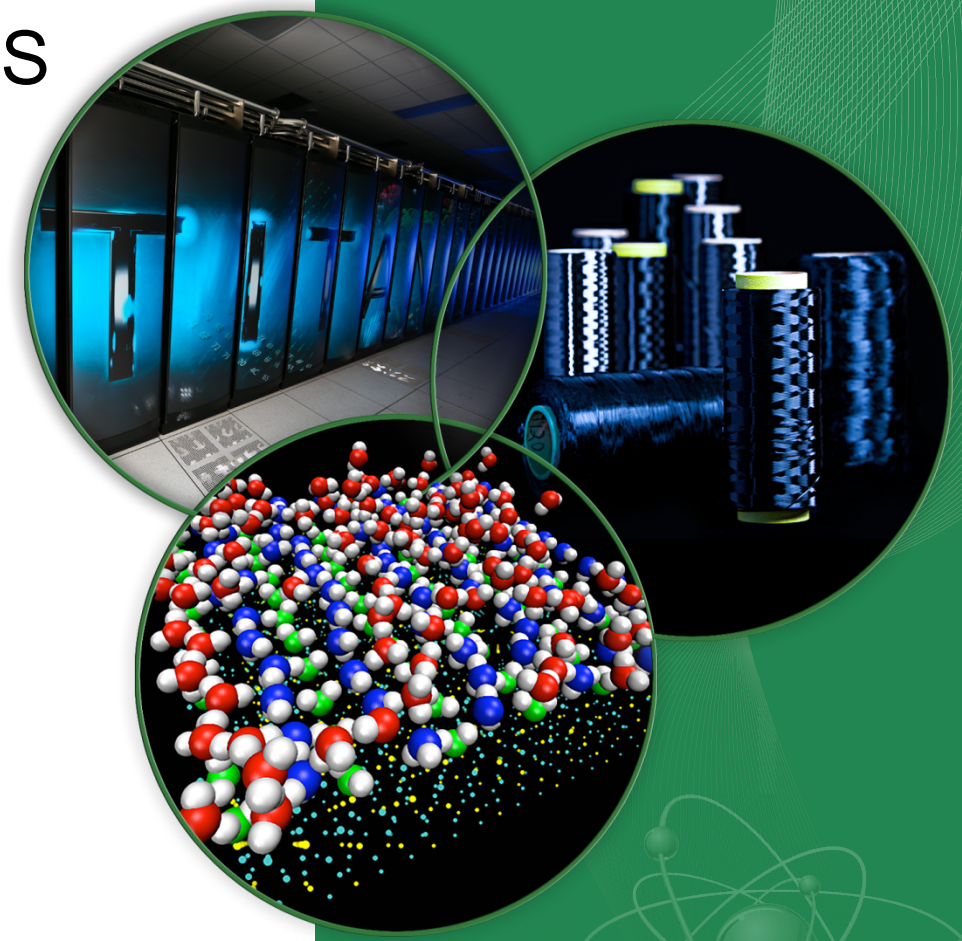
Data Staging and Asynchronous I/O in ADIOS

Hasan Abbasi	**ORNL**
Jong Choi	ORNL
Greg Eisenhauer	Georgia Tech
Scott Klasky	ORNL
Manish Parashar	Rutgers
Norbert Podhorszki	ORNL
Nagiza Samatova	NCSU
Karsten Schwan	Georgia Tech
Matthew Wolf	Georgia Tech



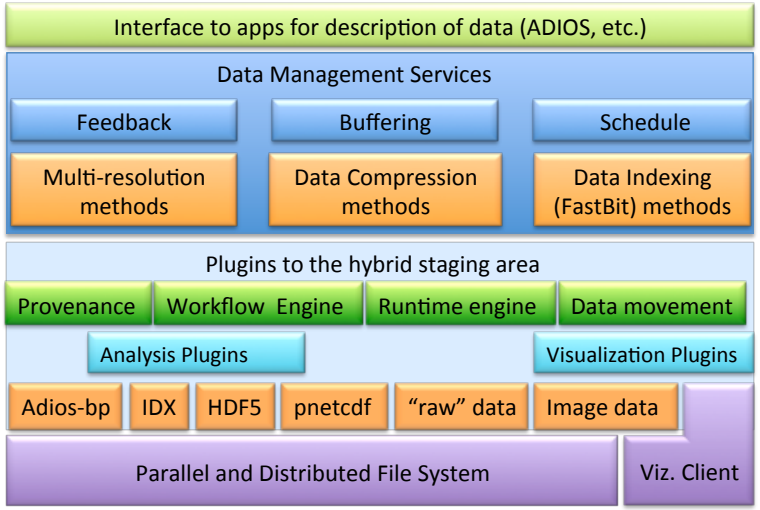
Outline

- ADIOS Overview
- Introduction to Staging
- Data Management in I/O Pipelines
- Staging in ADIOS
- Network and System service discussion

ADIOS

http://www.nccs.gov/user-support/center-projects/adios/

- Abstracts Data-at-Rest to Data-in-Motion for HPC
  - Provides portable, fast, scalable, easy-to-use, metadata rich output
  - Dynamically allows users to change the method during an experiment/simulation
- Provides solutions for "90% of the applications"
- ADIOS has been cited almost 1,000 times



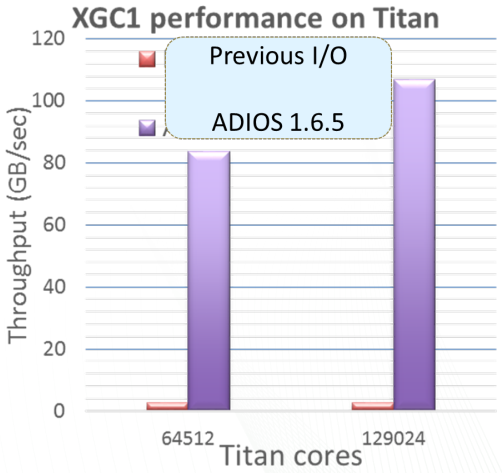
- Astrophysics
- Climate
- Combustion
- CFD
- Environmental Science
- Fusion
- Geoscience
- Materials Science
- Medical: Pathology
- Neutron Science
- Nuclear Science
- Quantum Turbulence
- Relativity
- Seismology
- Sub-surface modeling
- Weather

Improving I/O Methods for High End simulations

- Reduce I/O overhead, reduce network data movement, improve writing and reading performance
- To achieve this goal, ADIOS provides many methods
  - Posix (1 file per process, independent set of files)
  - Posix (1 file per process + metadata; read as one dataset)
  - MPI-Lustre (MPI-IO writing to 1 global file)
  - Aggregate (1 file per OST) + 1 metadata file
  - BG (1 file per rack) + 1 metadata file
  - ….
- There's no single right answer for all users.
  - ADIOS gives the user flexibility without rewriting code.

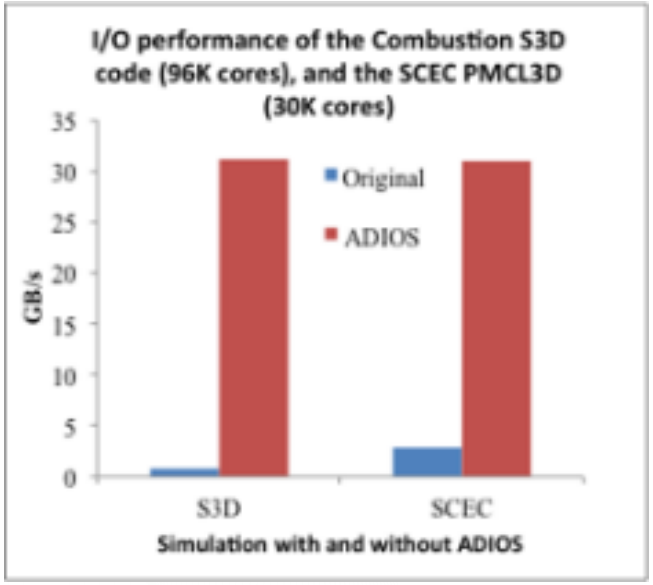
Large Writes Per Many cores

- First effort shows performance goes from 50 GB/s to over 100 GB/s
- New features for IBM BG/Q to eliminate the serial process in ADIOS for the metadata creation is now optional
  - Metadata creation is serial due to the problem of threadsafe MPI on most systems
- Testing has begun to use staging to write data
  - Problem is size of the staging area
    - Requires over 10K cores for staging…
    - GPU on staging is useless if we do NOT do other processing



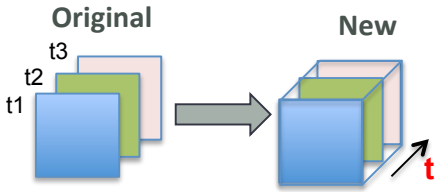
Small Writes per many cores (Combustion)

- Requires high performance I/O due to large output (200 GB/10 minutes)
- Frequent reading of large datasets on a small number of processors for analytics
- Individual process output is small, leading to low utilization of network bandwidth with other I/O solutions
- Reading of large datasets with a different access pattern than they were written out leads to
  - frequent seeking for data
  - very low read bandwidth
- Analysis codes spend 90% of their time reading data
- Allowed ADIOS team to focus on small but frequent output data

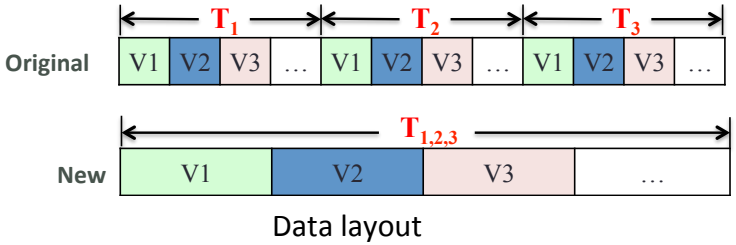


Spatial Temporal Aggregation

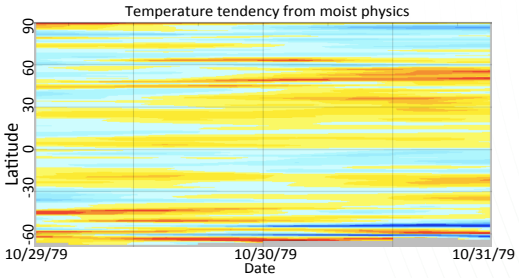
- Temporal aggregation is to open up another horizon to further consolidate data
- Data of multiple time steps are merged at each process
- Data is written out only at the last time step or reaches the boundary of memory capacity
- Achieved up to 70x speedup for read performance, and 11x speedup for write performance in mission critical climate simulation GEOS-5 (NASA), on Jaguar



A 2-D variable at 3 time steps are merged into a 3-D variable with time as new dimension

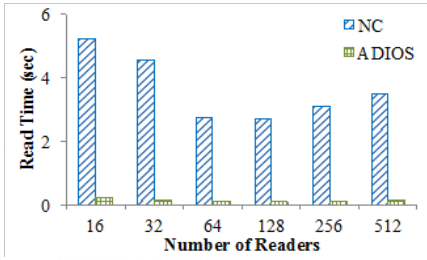


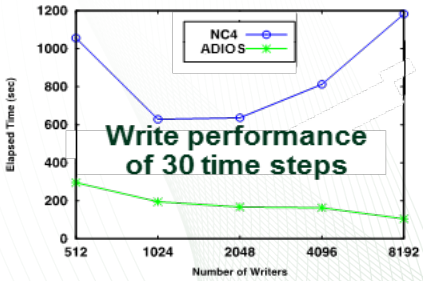
- Common read patterns for GEOS-5 users are reduced from 10 – 0.1 seconds
- Allows interactive data exploration for mission critical visualizations



GEOS-5 Results

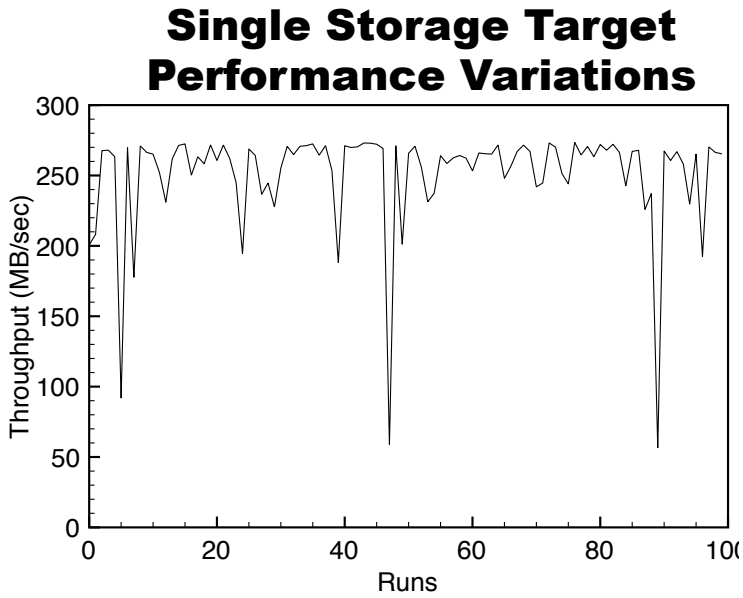
Read performance of a 2D slice of a 3D variable + time

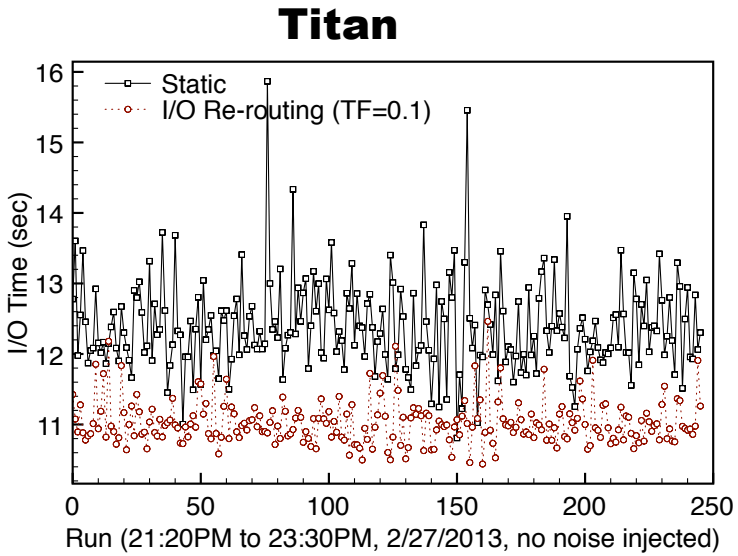


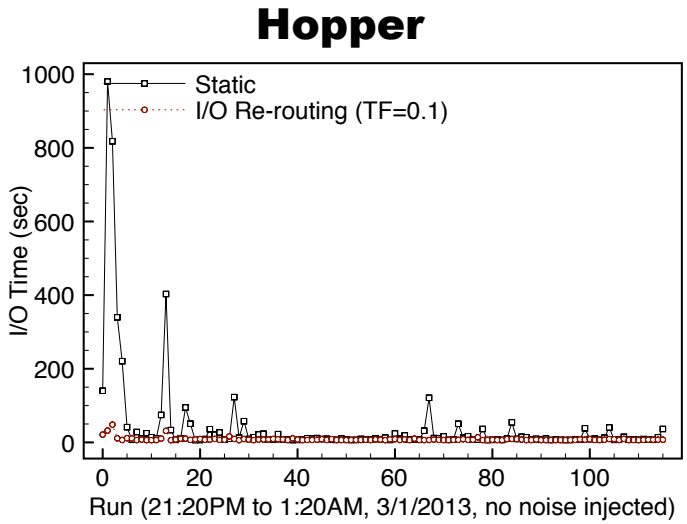


I/O Variability

- Problem
  - Techniques that achieved high performance I/O
    - Aggregation with write-behind strategy
    - Stripe alignment: to avoid contention
  - Are these techniques sufficient to get the peak I/O performance?

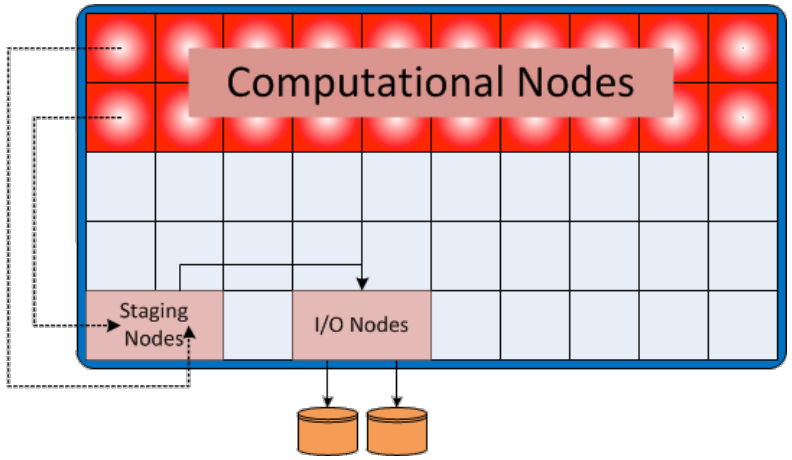






Introduction to Staging

- Initial development as a research effort to minimize I/O overhead
- Draws from past work on threaded I/O
- Exploits network hardware support for fast data transfer to remote memory

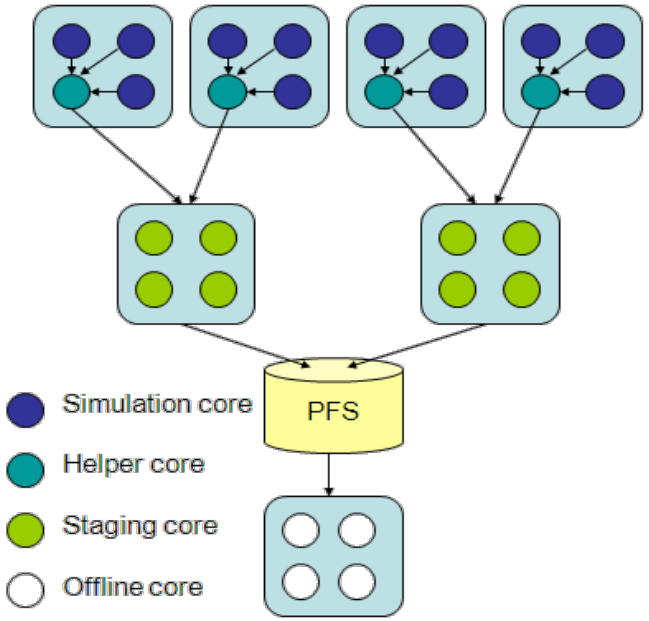


Hasan Abbasi, Matthew Wolf, Greg Eisenhauer, Scott Klasky, Karsten Schwan, Fang Zheng: DataStager: scalable data staging services for petascale applications. Cluster Computing 13(3): 277-290 (2010)

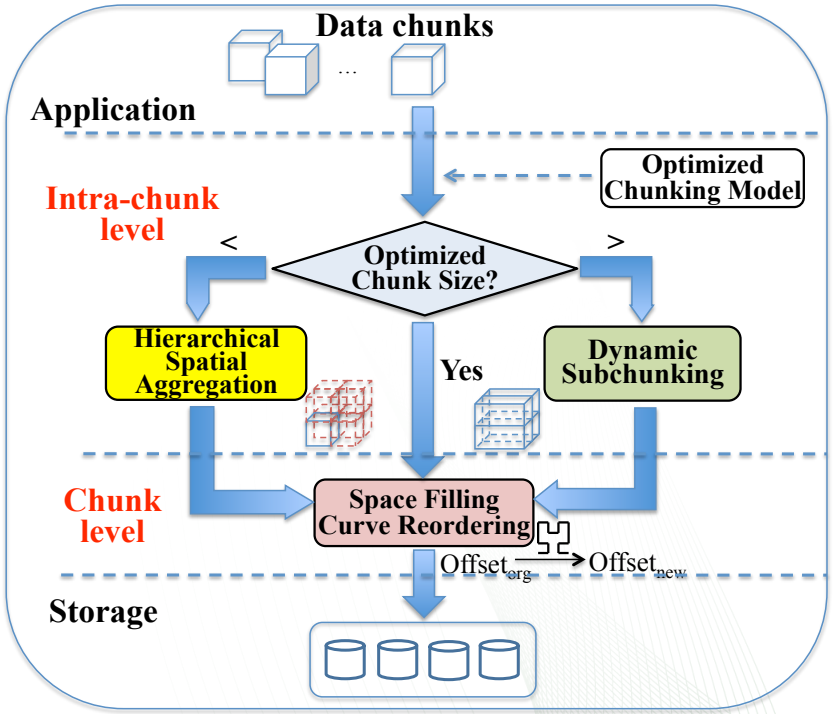
Ciprian Docan, Manish Parashar, Scott Klasky: DataSpaces: an interaction and coordination framework for coupled simulation workflows. Cluster Computing 15(2): 163-181 (2012)

Data Management in I/O Pipelines

- Perform computation in the *right* location
- Support dynamic placement
- Use data reduction techniques

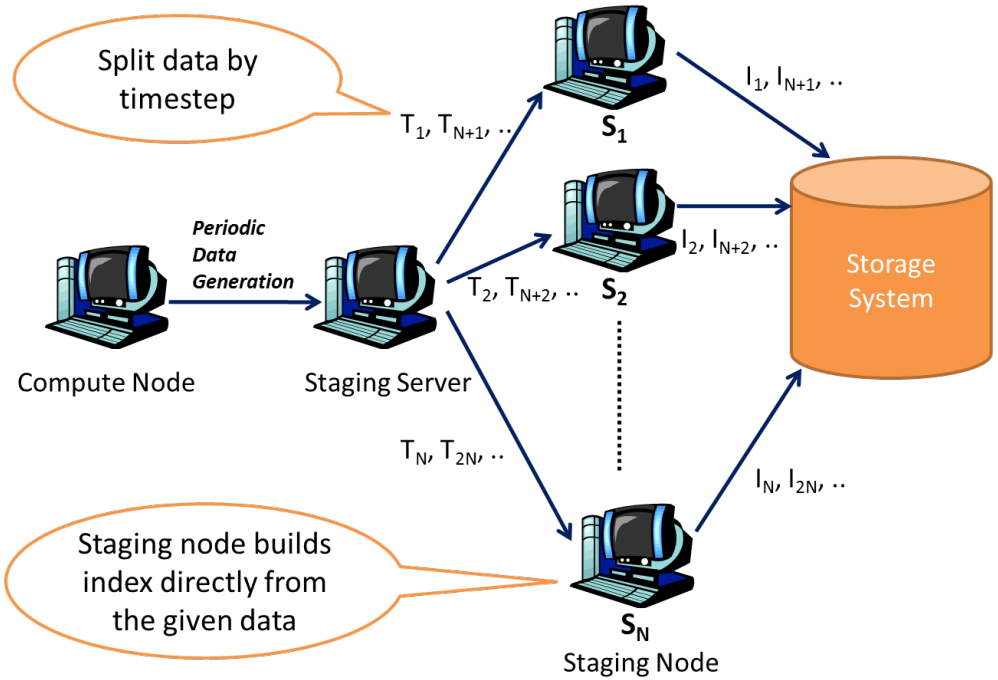


- Aggregation and chunking to improve data access
- End-to-End approach to data management



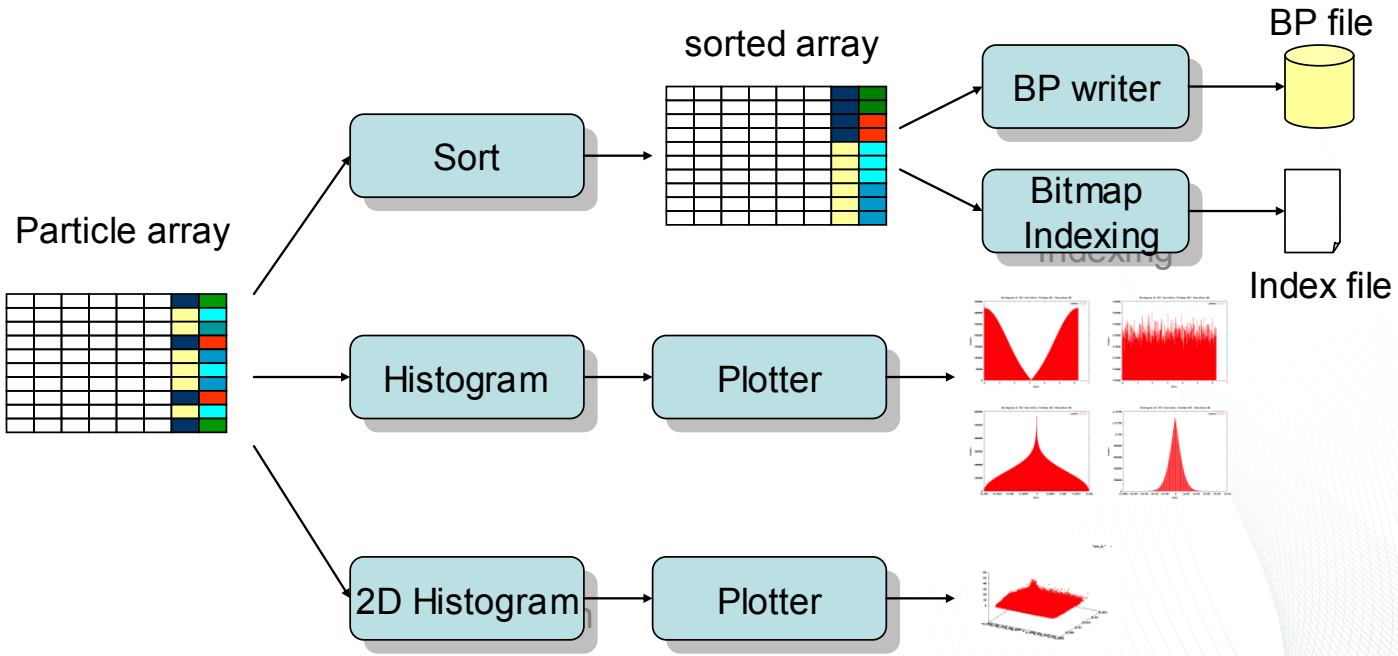
Indexing and Compression

- Extreme scale data enhancement and reduction
- Utilize in transit and in situ mechanisms
- Scientific compression schemes (ISABELA and ISOBAR)
- In situ indexing to enable fast query and data access
- Deployed as services in the pipeline

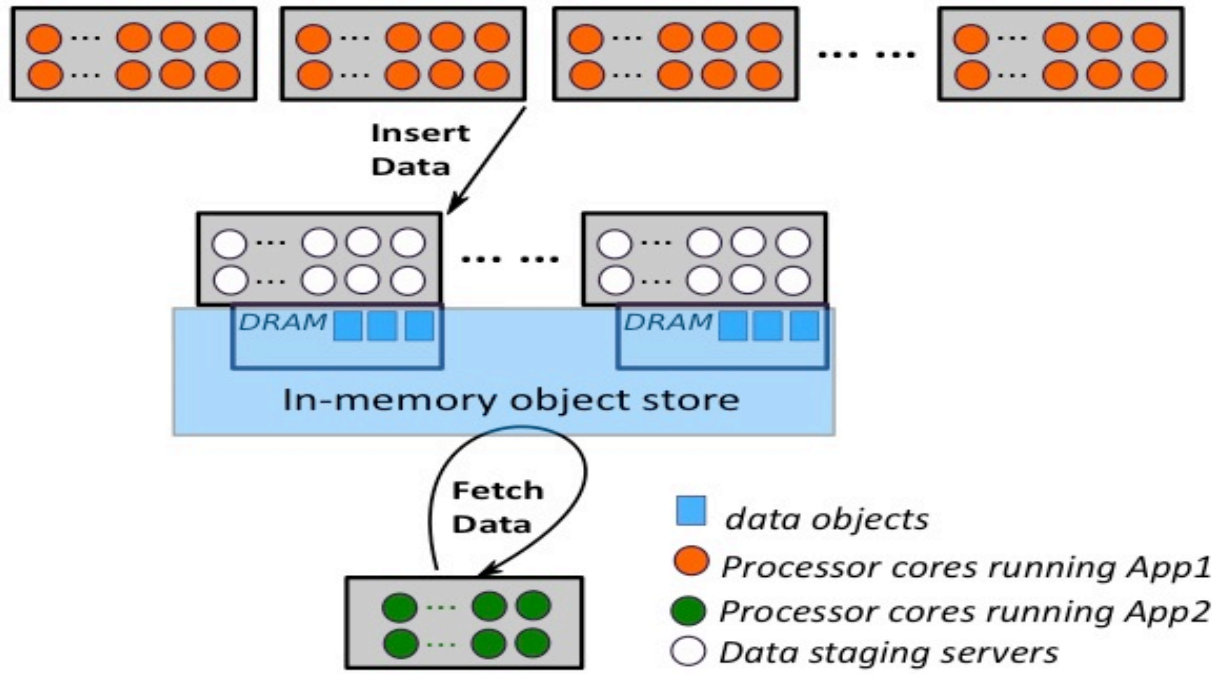


Predata: I/O Pipelines

- Use the staging nodes and create a workflow in the staging nodes.
- Allows us to explore many research aspects.
- Improve total simulation time by 2.7%
- Allow the ability to generate online insights into the 260GB data being output from 16,384 compute cores in 40 seconds.



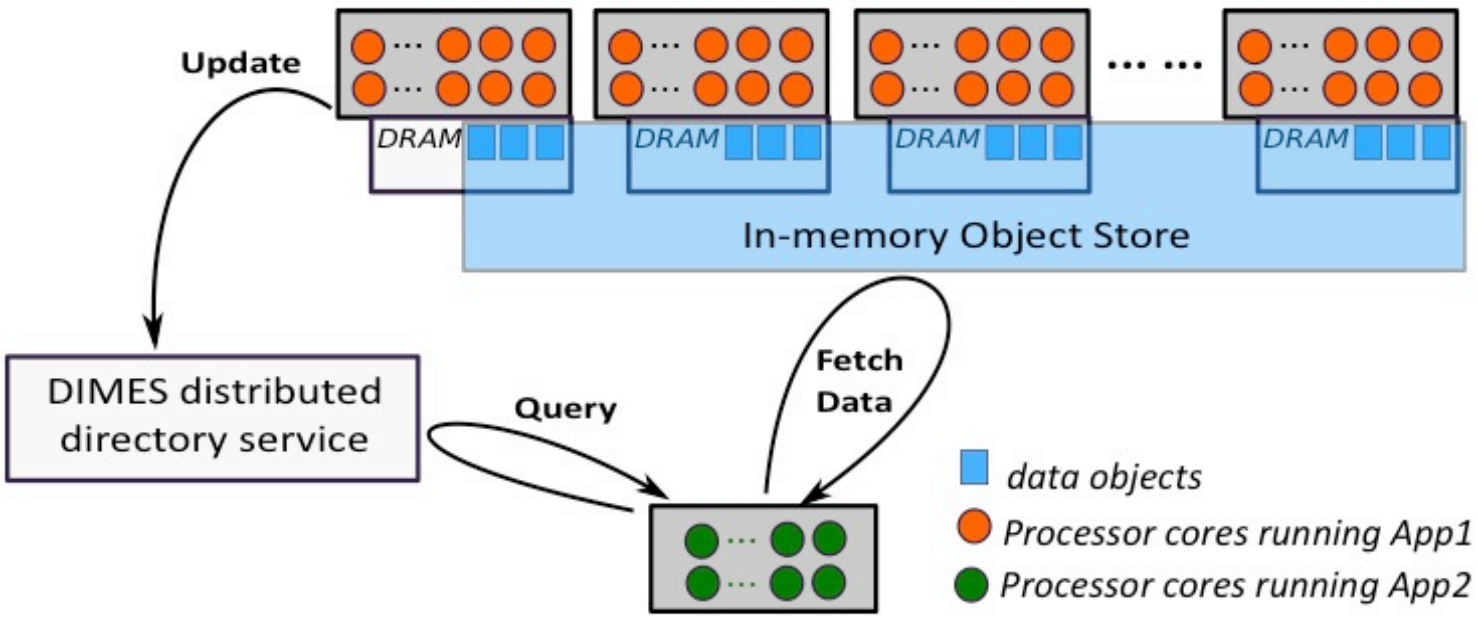
In-Memory Data Staging with DataSpaces



Staging-based (ADIOS DATASPACES transport method)

- Extract data from running simulations into the memory of staging servers
- Enables more loosely coupled data interactions
- Reduced resource contention, e.g., on-node memory

In-Memory Data Staging with DIMES

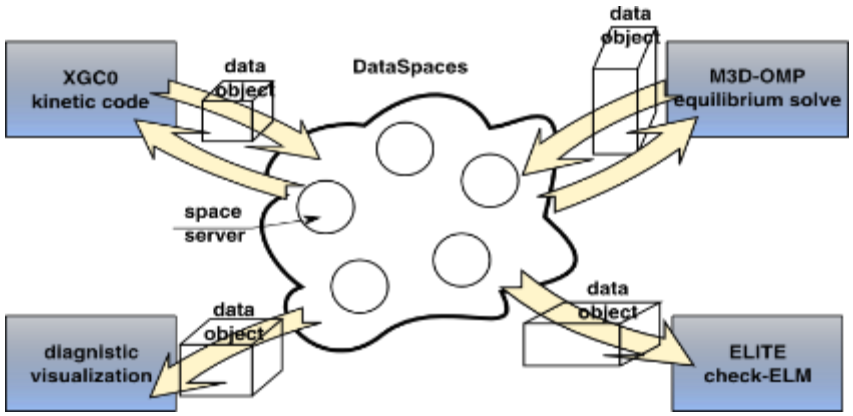


DIMES (ADIOS DIMES transport method)

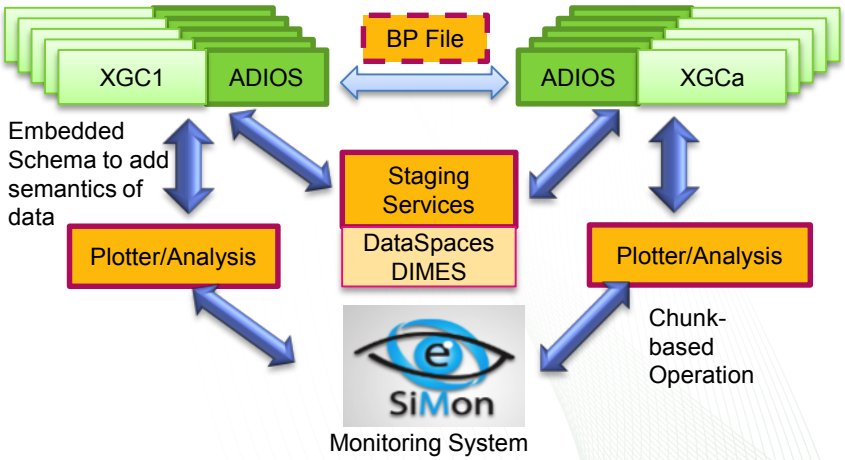
- Extract data from running simulations directly into another application's memory space
- Enable more tightly coupled data interactions
- Reduced network data movement (as compared to staging)

Application Coupling

Loose coupling of XGC0 and M3D-OMP and ELITE through ADIOS using DataSpaces

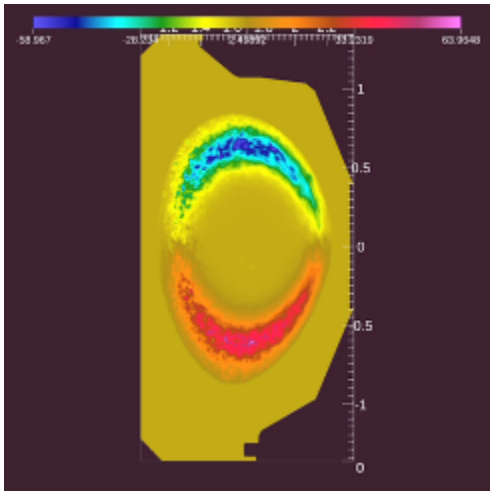


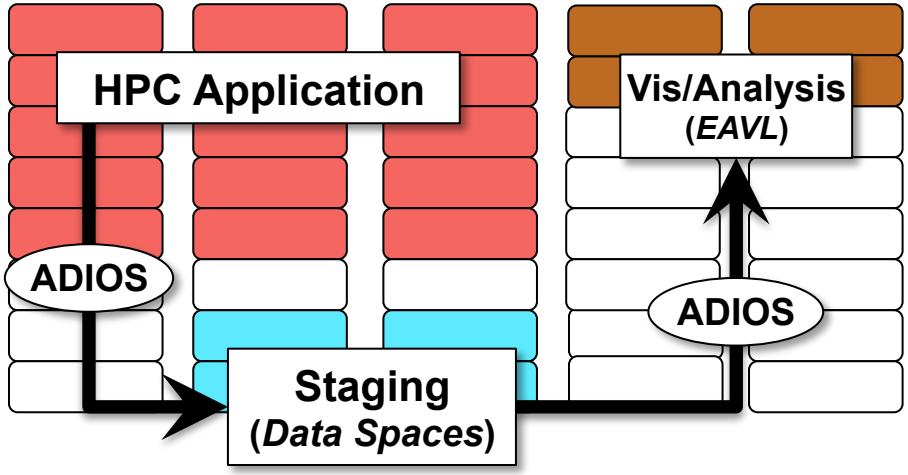
Tight coupling of XGC1 and XGCa in combination with a monitoring dashboard

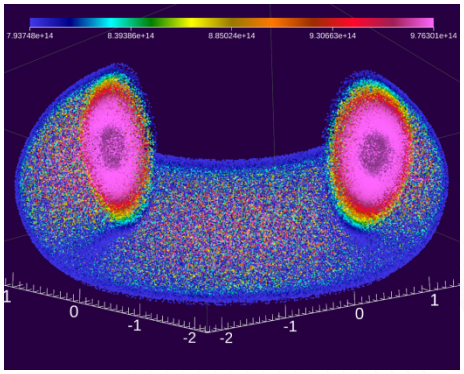


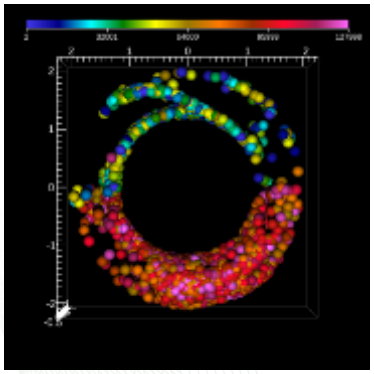
Loosely Coupled in-situ Visualization

- Visualization and application are de-coupled
- Uses ADIOS on Dataspaces
- Viz operations and rendering performed on system nodes
- XGC SciDAC simulation example

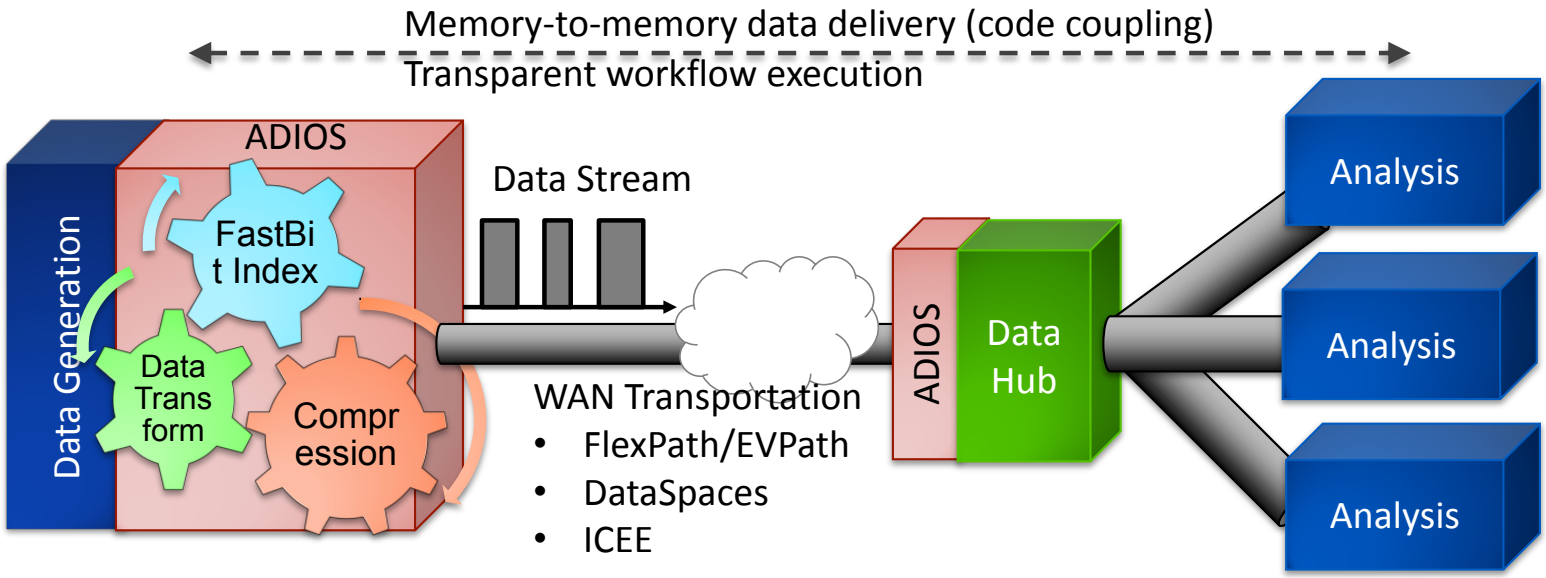








ICEE Method Enables WAN Staging



- Streaming data from experimental sources
- Transparent abstraction for moving data over LAN or WAN
- Integration of indexing/querying to minimize long distance data movement
- Multiple methods for moving data (ICEE, DataSpaces, Flexpath)

Network Services

- Managing Contention
  - We need to avoid resource contention to minimize I/O overhead on application performance
- Priority based flow control
  - Partitioning of streams into data and control will improve scalability
- Multicast for RDMA
  - A single source can be feeding data to multiple consumers for data staging operations
- Select/Callback support
  - Simple mechanism to check availability of data, particularly useful in combination with 1-sided communication

System Services

- Fault Tolerance
  - I/O services can add their own resiliency but need appropriate notification from the network layer to initiate recovery
- Scatter-Gather
  - More consistent support for scatter-gather (iovecs) across platforms
- RDMA to NVRAM
  - Particularly important for Summit
- Rendezvous and discovery
  - Identify I/O services, initiate and manage connections
  - Support for WANs and LANs in a similar API
- Feedback/progress APIs for RDMA operations
  - Services that augment the data stream to improve performance need to keep track of performance