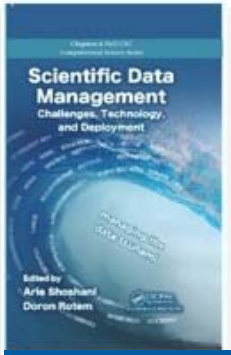
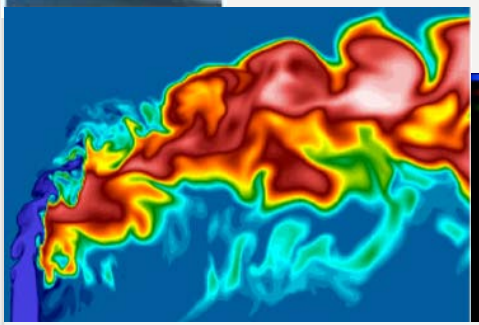
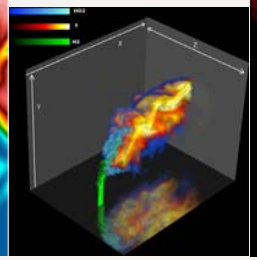
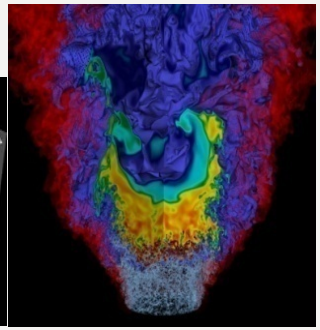
ADIOS 2.0: Why, What, When, Where









Scott A. Klasky

klasky@ornl.gov

Work supported under:

ASCR : SDAV, CPES, SDM, Runtime Staging, SAP, OLCF, Exact

OFES : GPSC, GSEP

NSF : EAGER, RDAV

NASA : ROSES

H. Abbasi[1], J. Choi[2], S. Ethier[8], R. Grout[7], Q. Liu[2], J. Logan[1], J. Lofstead[5], K. Moreland[5], M. Parashar[6], N. Podhorszki[1], D. Pugmire[1], N. Samatova[10], K. Schwan[4], A. Shoshani[3], Y. Tian[9], R. Vatsavai[1], M. Wolf[4], J. Wu[3], W. Yu[9]

[1]ORNL, [2] U.T. Knoxville, [3]LBNL, [4]Georgia Tech, [5]Sandia Labs, [6] Rutgers, [7]NREL, [8]PPPL, [9]Auburn University, [10]NCSU

Outline

- Requirements for an I/O system
  - Why not just make a better file system?
  - Why not just make a better visualization system?
  - Why not just make a better scientific application?
- ADIOS 101
  - What is ADIOS
  - ADIOS performance
  - ADIOS examples
- ADIOS 2
  - Service Oriented Architecture
  - Staging
  - Moving work to data
- Next Steps

Office of Science and Technology Policy
Executive Office of the President
New Executive Office Building
Washington, DC 20502

FOR IMMEDIATE RELEASE
March 29, 2012

Contact: Rick Weiss 202 456-6037 rweiss@ostp.eop.gov
Lisa-Joy Zgorski 703 292-8311 lisajoy@nsf.gov

OBAMA ADMINISTRATION UNVEILS "BIG DATA" INITIATIVE: ANNOUNCES $200 MILLION IN NEW R&D INVESTMENTS

Aiming to make the most of the fast-growing volume of digital data, the Obama Administration today announced a "Big Data Research and Development Initiative." By improving our ability to extract knowledge and insights from large and complex collections of digital data, the initiative promises to help solve some the Nation's most pressing challenges.

To launch the initiative, six Federal departments and agencies today announced more than $200 million in new commitments that, together, promise to greatly improve the tools and techniques needed to access, organize, and glean discoveries from huge volumes of digital data.

"In the same way that past Federal investments in information-technology R&D led to dramatic advances in supercomputing and the creation of the Internet, the initiative we are launching today promises to transform our ability to use Big Data for scientific discovery, environmental and biomedical research, education, and national security," said Dr. John P. Holdren, Assistant to the President and Director of the White House Office of Science and Technology Policy.

To make the most of this opportunity, the White House Office of Science and Technology Policy (OSTP)—in concert with several Federal departments and agencies—created the Big Data Research and Development Initiative to:

- Advance state-of-the-art core technologies needed to collect, store, preserve, manage, analyze, and share huge quantities of data.
- Harness these technologies to accelerate the pace of discovery in science and engineering, strengthen our national security, and transform teaching and learning; and
- Expand the workforce needed to develop and use Big Data technologies.

Today's initiative responds to recommendations by the President's Council of Advisors on Science and Technology, which last year concluded that the Federal Government is under-investing in technologies related to Big Data. In response, OSTP launched a Senior Steering Group on Big Data to coordinate and expand the Government's investments in this critical area. Today's announcement describes the first wave of agency commitments to support this initiative, including:

National Science Foundation and the National Institutes of Health - Core Techniques and Technologies for Advancing Big Data Science & Engineering
"Big Data" is a new joint solicitation supported by the National Science Foundation (NSF) and the National Institutes of Health (NIH) that will advance the core scientific and technological means of managing, analyzing, visualizing, and extracting useful information from large and diverse data sets. This will accelerate scientific discovery and lead to new fields of inquiry that would otherwise not be possible. NIH is particularly interested in imaging, molecular, cellular, electrophysiological, chemical, behavioral, epidemiological, clinical, and other data sets related to health and disease.

In addition to its funding of the Big Data solicitation, NSF is also:

- Encouraging research universities to develop interdisciplinary graduate programs to prepare the next generation of data scientists and engineers;
- Funding a $10 million project based at the University of California, Berkeley, that will integrate three powerful approaches for turning data into information - machine learning, cloud computing, and crowd sourcing;
- Providing the first round of grants to support "EarthCube" – a system that will allow geoscientists to access, analyze and share information about our planet;
- Issuing a $2 million award for a research training group to support training for undergraduates to use graphical and visualization techniques for complex data.
- Providing $1.4 million in support for a focused research group of statisticians and biologists to tell us about protein structures and biological pathways.
- Convening researchers across disciplines to determine how Big Data can transform teaching and learning.

Department of Defense – Data to Decisions: The Department of Defense (DoD) is "placing a big bet on big data" investing $250 million annually (with $60 million available for new research projects) across the Military Departments in a series of programs that will:

- Harness and utilize massive data in new ways and bring together sensing, perception and decision support to make truly autonomous systems that can maneuver and make decisions on their own.
- Improve situational awareness to help warfighters and analysts and provide increased support to operations. The Department is seeking a 100-fold increase in the ability of analysts to extract information from texts in any language, and a

similar increase in the number of objects, activities, and events that an analyst can observe.

To accelerate innovation in Big Data that meets these and other requirements, DoD will announce a series of open prize competitions over the next several months.

In addition, the Defense Advanced Research Projects Agency (DARPA) is beginning the XDATA program, which intends to invest approximately $25 million annually to develop computational techniques and software tools for analyzing large volumes of data, both semi-structured (e.g., tabular, relational, categorical, meta-data) and unstructured (e.g., text documents, message traffic). Central challenges to be addressed include:

- Developing scalable algorithms for processing imperfect data in distributed data stores; and
- Creating effective human-computer interaction tools for facilitating rapidly customizable visual reasoning for diverse missions.

The XDATA program will support open source software toolkits to enable flexible software development for users to process large volumes of data in timelines commensurate with mission workflows of targeted defense applications.

National Institutes of Health – 1,000 Genomes Project Data Available on Cloud: The National Institutes of Health is announcing that the world's largest set of data on human genetic variation – produced by the international 1000 Genomes Project – is now freely available on the Amazon Web Services (AWS) computing cloud. At 200 terabytes – the equivalent of 16 million file cabinets filled with text, or more than 30,000 standard DVDs – the current 1000 Genomes Project data set is a prime example of big data, where data sets become so massive that few researchers have the computing power to make best use of them. AWS is hosting the 1000 Genomes Project as a publically available data set for free and researchers only will pay for the computing services that they use.

Department of Energy – Scientific Discovery Through Advanced Computing: As part of its Scientific Discovery through Advanced Computing program, the Department of Energy will provide $25 million in funding for the Scalable Data Management, Analysis and Visualization Institute. Led by Lawrence Berkeley National Laboratory, the Institute will bring together the expertise of six National Laboratories and seven universities, with the goal of developing new and improved tools to help scientists manage and visualize data. The need for these new tools has grown as the simulations running on the Department of Energy's supercomputers have increased in size and complexity.

US Geological Survey – Big Data for Earth System Science: USGS is announcing the latest awardees for grants it issues through its John Wesley Powell Center for Analysis and Synthesis. The Center catalyzes innovative thinking in Earth system science by providing scientists a place and time for in-depth analysis, state-of-the-art computing capabilities, and collaborative tools invaluable for making sense of huge data

sets. These Big Data projects will improve our understanding of issues such as species response to climate change, earthquake recurrence rates, and the next generation of ecological indicators.

Further details about each department's or agency's commitments can be found at the following websites by 2 pm today:

NSF: http://www.nsf.gov/news/news_summ.jsp?cntn_id=123607

HHS/NIH: http://www.nih.gov/news/health/mar2012/nhgri-29.htm

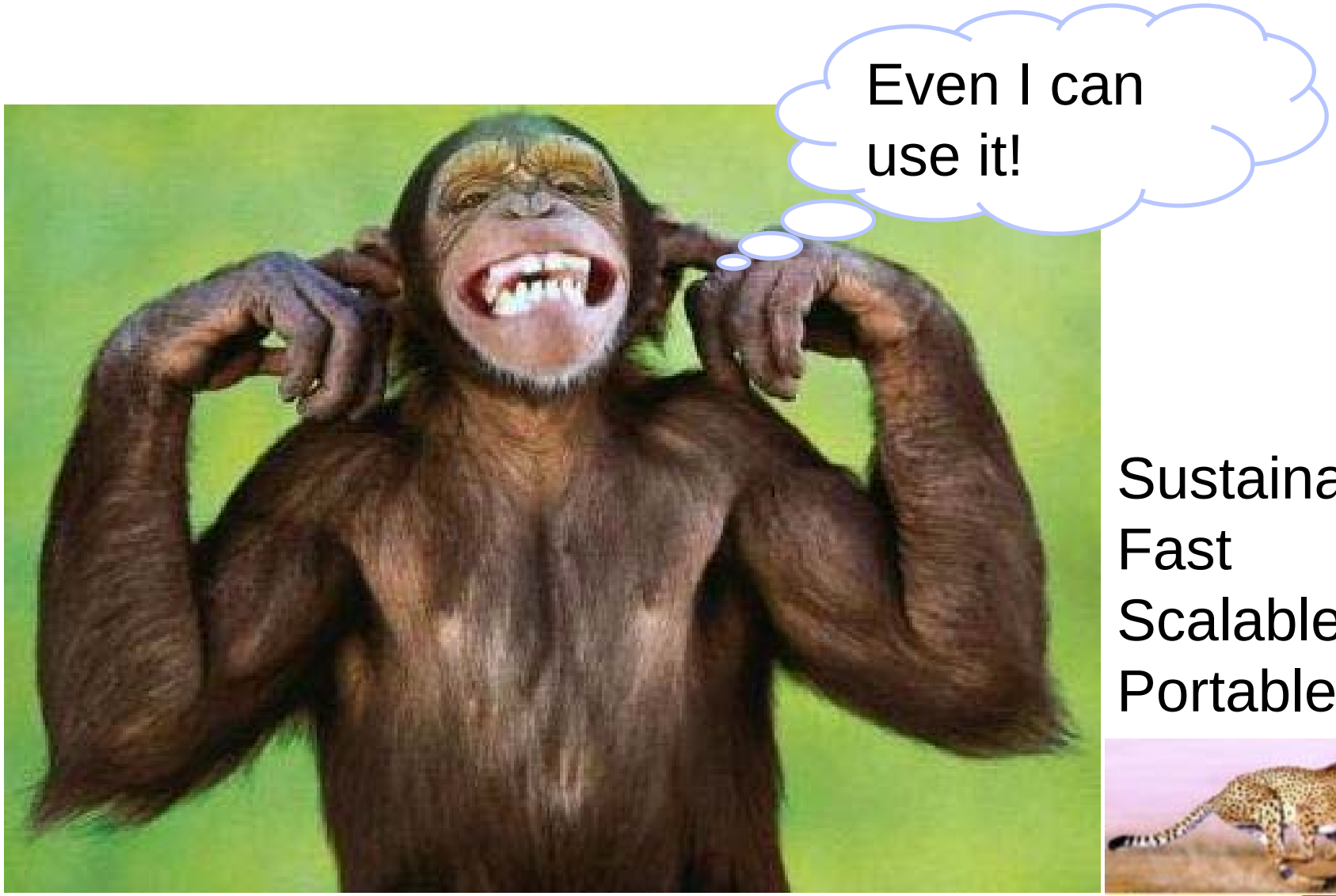
DOD: www.DefenseInnovationMarketplace.mil

DARPA: http://www.darpa.mil/NewsEvents/Releases/2012/03/29.aspx

USGS: http://powellcenter.usgs.gov

For more information on OSTP, visit whitehouse.gov/ostp

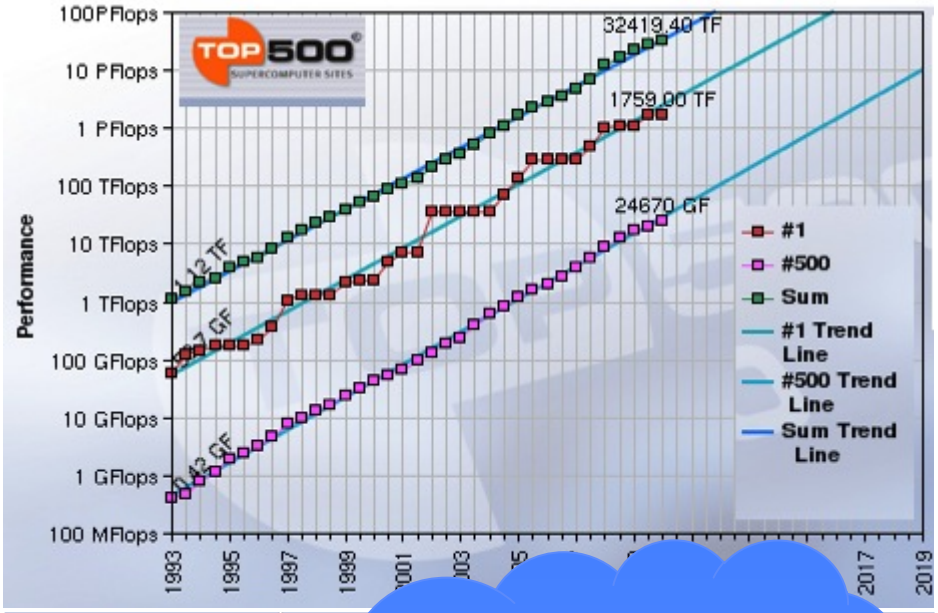
We want our system to be so easy, even a chimp can use it



Sustainable
Fast
Scalable
Portable

Extreme scale computing

- Trends
  - More FLOPS
  - Limited number of users at the extreme scale
- Problems
  - Performance
  - Resiliency
  - Debugging
  - Getting Science done
- Problems will get worse
  - Need a "revolutionary" way to store, access, debug to get the science done!
- ASCI purple (49 TB/140 GB/s) – JaguarPF (300 TB/200 GB/s)



Systems	20			
System Peak Flops/'s	2			
System Memory				
Node Performance				
Node Memory BW				/s
Node Concurrency				(1000)
Interconnect BW	1.5 GB			50 GB/s
System Size (Nodes)	18,700	100,0	,000	O(Million)
Total Concurrency	225,000	3 Million	50 Million	O(Billion)
Storage	15 PB	30 PB	150 PB	300 PB
I/O	0.2 TB/s	2 TB/s	10 TB/s	20 TB/s
MTTI	Days	Days	Days	O(1Day)
Power	6 MW	~10 MW	~10 MW	~20 MW

ILLUSTRATION: A. TOVEY

Most people get < 5 GB/s at scale

From J. Dongarra, "Impact of Architecture and Technology for Extreme Scale on Software and Algorithm Design," Cross-cutting Technologies for Computing at the Exascale, February 2-5, 2010.

Next generation I/O and file system challenges

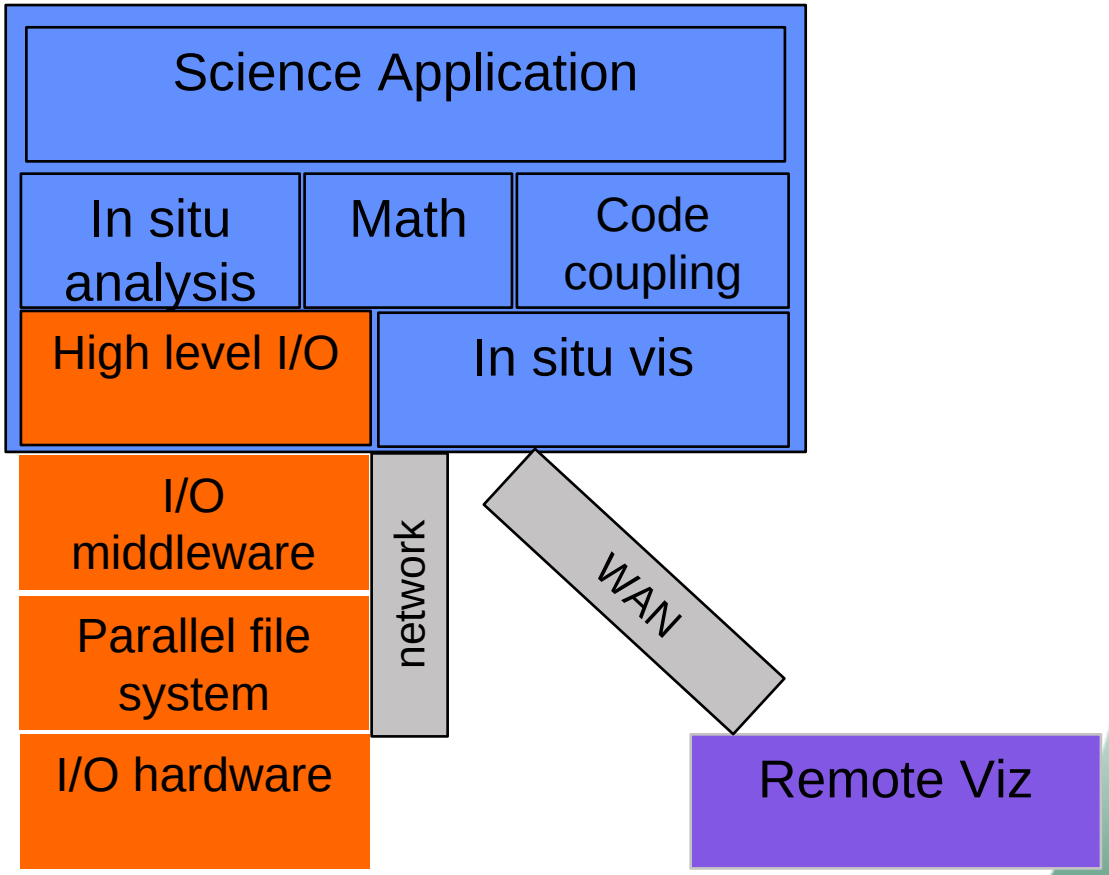
- At the architecture or node level
  - Use increasingly deep memory hierarchies coupled with new memory properties
- At the system level
  - Cope with I/O rates and volumes that stress the interconnect and can severely limit application performance
  - Can consume unsustainable levels of power
- At the exascale
  - Immense aggregate I/O needs with potentially uneven loads placed on underlying resource
  - Can result in data hotspots, interconnect congestion and similar issues

File Systems

File system	Hard links	Symbolic links	Block journaling	Metadata-only journaling	Case-sensitive	Case-preserving	File Change Log	Snapshot	XIP	Encryption	COW	integrated LVM	Data deduplication	Volumes are resizeable
CP/M file system	No	No	No	No	No	No	No	No	No	No	No	No	No	Unknown
DECtape	No	No	No	No	No	No	No	No	No	No	No	No	No	Unknown
Level-D	No	No	No	No	No	No	No	No	No	No	Unknown	Unknown	Unknown	Unknown
RT-11	No	No	No	No	No	No	No	No	No	No	No	No	No	Unknown
DOS (GEC)	No	No	No	No	No	No	No	No	No	No	No	No	No	Unknown
OS4000	No	Yes[71]	No	No	No	No	No	No	No	No	No	No	No	Unknown
V6FS	Yes	No	No	No	Yes	Yes	No	No	No	No	No	No	No	Unknown
V7FS	Yes	No[72]	No	No	Yes	Yes	No	No	No	No	No	No	No	Unknown
FAT12	No	No	No	No	No	Partial	No	No	No	No	No	No	No	Offline[73]
FAT16	No	No	No	No	No	Partial	No	No	No	No	No	No	No	Offline[73]
FAT32	No	No	No	No	No	Partial	No	No	No	No	No	No	No	Offline[73]
exFAT	No	No	Unknown	No	No	Yes	No	Unknown	Unknown	No	Unknown	Unknown	Unknown	Unknown
GFS	Yes	Yes[74]	Yes	Yes[75]	Yes	Yes	No	No	No	No	Unknown	Unknown	Unknown	Unknown
GPFS	Yes	Yes	Unknown	Unknown	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	No	Online
HPFS	No	No	No	No	No	Yes	No	Unknown	No	No	Unknown	Unknown	No	Unknown
NTFS	Yes	Yes[76]	No[77]	Yes[77]	Yes[78]	Yes	Yes	Partial[79]	Yes	Yes	Partial	Unknown	No	Online[80]
HFS	No	Yes[81]	No	No	No	Yes	No	No	No	No	No	No	No	Unknown
HFS Plus	Yes[82]	Yes	No	Yes[83]	Partial[84]	Yes	Yes[85]	No	No	No[86]	No	No	No	Offline
FFS	Yes	Yes	No	No[87]	Yes	Yes	No	No	No	No	No	No	No	Offline (cannot be shrunk)[88]
UFS1	Yes	Yes	No	No	Yes	Yes	No	No	No	No	No	No	No	Unknown
UFS2	Yes	Yes	No	No[89][90]	Yes	Yes	No	Yes	Unknown	No	No	No	No	Offline (cannot be shrunk)[91]
LFS	Yes	Yes	Yes[92]	No	Yes	Yes	No	Yes	No	No	Unknown	Unknown	Unknown	Unknown
ext2	Yes	Yes	No	No	Yes	Yes	No	No	Yes[93]	No	No	No	No	Online[94]
ext3	Yes	Yes	Yes[95]	Yes	Yes	Yes	No	No	Yes	No	No	No	No	Online[94]
ext4	Yes	Yes	Yes[95]	Yes	Yes	Yes	No	No	Yes	No	No	No	No	Online[94]
Lustre	Yes	Yes	Yes[95]	Yes	Yes	Yes	Yes in 2.0	No[58]	No	No[58]	No[58]	No[58]	No[58]	Online[96]
NILFS	Yes	Yes	Yes[92]	No	Yes	Yes	Yes	Yes	No	No	Yes	Unknown	Unknown	Unknown
ReiserFS	Yes	Yes	No[97]	Yes	Yes	Yes	No	No	No	No	No	No	No	Offline
Reiser4	Yes	Yes	Yes	No	Yes	Yes	No	Unknown	No	Yes[98]	Yes	No	Unknown	Online (can only be shrunk offline)
OCFS	No	Yes	No	No	Yes	Yes	No	No	No	No	Unknown	Unknown	Unknown	Unknown
OCFS2	Yes	Yes	Yes	Yes	Yes	Yes	No	Partial[99]	No	No	Unknown	No	No	Online for version 1.4 and higher
Reliance	No	No	No[100]	No	No	Yes	No	No	No	No	Yes	No	No	Unknown
Reliance Nitro	Yes	Yes	No[100]	No	Depends on OS	Yes	No	No	No	No	Yes	No	No	Unknown
XFS	Yes	Yes	Yes	Yes	Yes[101]	Yes	No	No	No	No	No	No	No	Online (cannot be shrunk)
JFS	Yes	Yes	No	Yes	Yes[102]	Yes	No	Yes	No	No	Yes	Unknown	Unknown	Online (cannot be shrunk)[103]
QFS	Yes	Yes	No	Yes	Yes	Yes	No	No	No	No	Unknown	Unknown	Unknown	Unknown
Be File System	Yes	Yes	No	Yes	Yes	Yes	Unknown	No	No	No	No	No	No	Unknown
NSS	Yes	Yes	Unknown	Yes	Yes[104]	Yes[104]	Yes[105]	Yes	No	Yes	Unknown	Unknown	Unknown	Unknown
NWFS	Yes[106]	Yes[106]	No	No	Yes[104]	Yes[104]	Yes[105]	Unknown	No	No	No	Yes[107]	Unknown	Unknown
ODS-2	Yes	Yes[108]	No	Yes	No	No	Yes	Yes	No	No	Unknown	Unknown	Unknown	Unknown
ODS-5	Yes	Yes[108]	No	Yes	No	Yes	Yes	Yes	Unknown	No	Unknown	Unknown	Unknown	Unknown
UDF	Yes	Yes	Yes[92]	Yes[92]	Yes	Yes	No	No	Yes	No	No	No	No	Unknown
VxFS	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes[109]	Unknown	No	Unknown	Unknown	Unknown	Unknown
Fossil	No	No	No	No	Yes	Yes	Yes	Yes	No	No	Unknown	No	Yes[110]	Unknown
ZFS	Yes	Yes	Yes[111]	No[111]	Yes	Yes	No	Yes	No	Yes	Yes	Yes	Yes	Online (cannot be shrunk)[112]
VMFS2	Yes	Yes	No	Yes	Yes	Yes	No	No	No	No	Unknown	Unknown	Unknown	Unknown
VMFS3	Yes	Yes	No	Yes	Yes	Yes	No	No	No	No	Unknown	Unknown	Unknown	Unknown
Btrfs	Yes	Yes	Unknown	Yes	Yes	Yes	Unknown	Yes	No	Planned[113]	Yes	Yes	Work-in-Progress	Online
File system	Hard links	Symbolic links	Block journaling	Metadata-only journaling	Case-sensitive	Case-preserving	File Change Log	Snapshotting	XIP	Encryption	COW	integrated LVM	Data deduplication	Volumes are resizeable

Tools and Technologies to work with large data

- MPI
- ROMIO
- HDF5
- Netcdf-4
- Globus on-line
- SciDB
- Paraview
- Visit
- …



But ….

- We want to view I/O as not just “I/O” but rather
  - I/O pipelines
- They need semantic knowledge
  - It’s just not a bunch of bytes
  - How do you interpret the information to analyze it? To visualize it?
- This tells us that we need self-describing files

But ….

- Data is getting large, and produced from many sources (MPI procs, sensors, …)
- Data can be analyzed, and visualized
  - In-situ
  - In-transit
  - Co-processing
  - On-clusters
  - Clouds
  - Desktops
  - Smart-phones

So…

- We need self-describing data streams(chunks) from many **process**es which can
  - Have embedded code
  - Have semantic knowledge of how to work with it
  - Have embedded workflows
  - Embed provenance information

And

- Data naturally comes in **group**s
  - Check-point restart data
  - Analysis data
  - Visualization data
  - Monitoring data

But how do we make this easy-to-use

- We need to create schema's (standards) that doesn't involve much user involvement
  - For in-transit visualization, analysis
  - In-situ visualization, analysis
  - Co-processing visualization, analysis
  - For code-coupling
  - For any operations on the data which do NOT require human intervention

Development

- We want to have different systems people build different systems layer
  - Viz people build the viz. layer
  - Analysis people build the analysis layer
  - …
- Ideally we need to create "teams" to create the software
  - Teams for the apps
  - Teams for the analysis
  - Teams for the math
  - Teams for data management
  - Teams for visualization

But…

- All of these teams have a diverse set of developers
- All of these components need to be created independently
- Debugged and Tested independently
- All of these components need to be stand-alone, and easily integrated in the complex system
  - On 1 machine
  - Integrated on many machines
- And of of these components need to work together in one (or many small) workflows

Moving towards ADIOS 2.0 (Nomenclature)

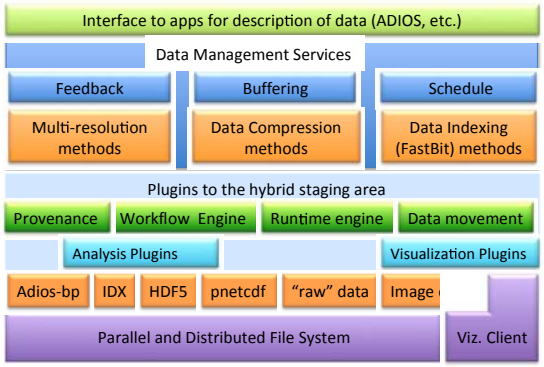
- Staging Area
  - Set of resources that serves as a “staging” point for data
  - Combines compute with memory
- DataSpace
  - A staging model for ADIOS that provides a global description of data and allows distributed access
- Hybrid Staging
  - Abstracts staging further, allowing cores on a compute node and extra nodes to run SDMA services
- Service Oriented Architecture
  - Mix and match components of a pipeline dynamically, as separate exacutables
- Data pipeline
  - Our vision of how extreme scale data processing will happen in the future
  - Already important at petascale

Thus: the SOA philosophy

- The overarching design philosophy of our framework is based on the Service-Oriented Architecture
  - Used to deal with system/application complexity, rapidly changing requirements, evolving target platforms, and diverse teams
- Applications constructed by assembling services based on a universal view of their functionality using a well-defined API
- Service implementations can be changed easily
- Integrated simulation can be assembled using these services
- Manage complexity while maintaining performance/scalability
  - Complexity from the problem (complex physics)
  - Complexity from the codes and how they are
  - Complexity of underlying disruptive infrastructure
  - Complexity from coordination across codes and research teams

Adaptable I/O System

- Provides portable, fast, scalable, easy-to-use, metadata rich output with a simple API
- Change I/O method by changing XML input file
- Layered software architecture:
  - Allows plug-ins for different I/O implementations
  - Abstracts the API from the method used for I/O
- Open source: 3rd release: first beta in 2007
  - http://www.nccs.gov/user-support/center-projects/adios/
- High Writing Performance
  - S3D: 32 GB/s with 96K cores, 1.9MB/core: 0.6% I/O overhead with ADIOS
  - XGC1 code → 40 GB/s, SCEC code 30 GB/s
  - GTC code → 40 GB/s, GTS code: 35 GB/s



New applications of ADIOS

RAMGEN/Numeca

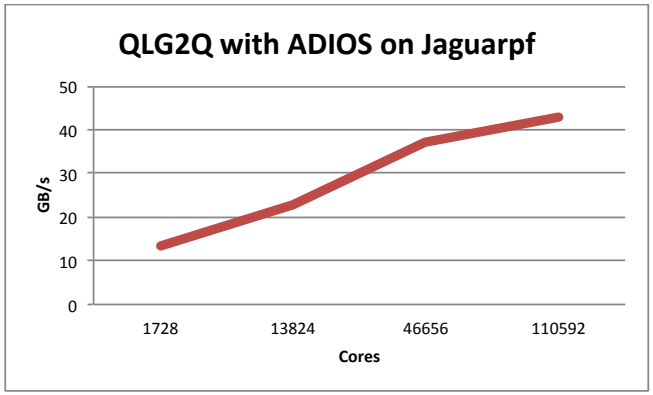
- CFD solver by Numeca International used by RAMGEN Power Systems at OLCF
- Two Body test case - 500 million grid cells - 3840 processes

Format	CGNS		BP	
Method	Partial IO	Buffered IO	MPI	MPI_LUSTER
Time [s]	2421	104	46	68
Scalability	0.04	1	2.27	1.53

- ADIOS BP format and parallel writing significantly accelerated from serial or aggregated CGNS output process
- ADIOS allowed for running simulation on 3.7 billion grid cells, which scale was not possible before

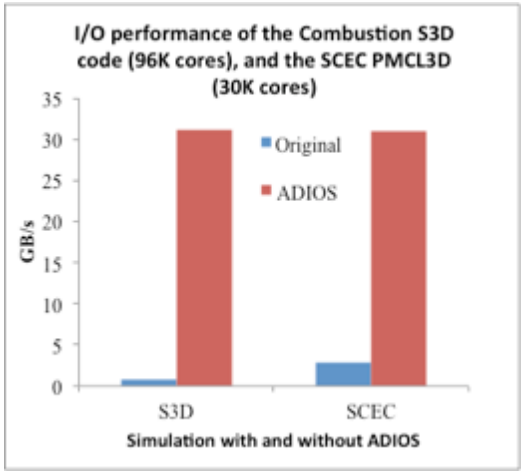
QLG2Q

- Two-qubit quantum lattice gas model for quantum turbulence by Min Soe, RSU
- about 30MB data per process (1.3TB@46k - 3.2TB@110k),
- Old I/O codes: MPI-IO with file views (single file) and POSIX (one file per core): **did not scale beyond 20+k**
- ADIOS: **30-40 GB/sec** write/read speed on Jaguarpf with 36^3/48^3 runs.

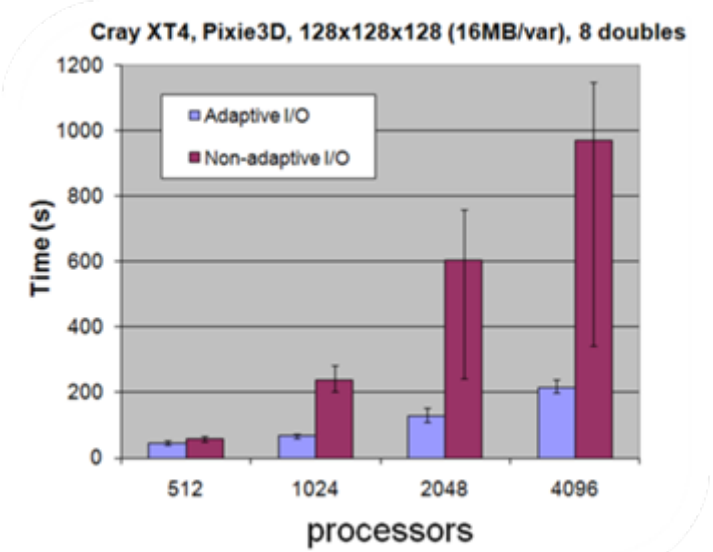


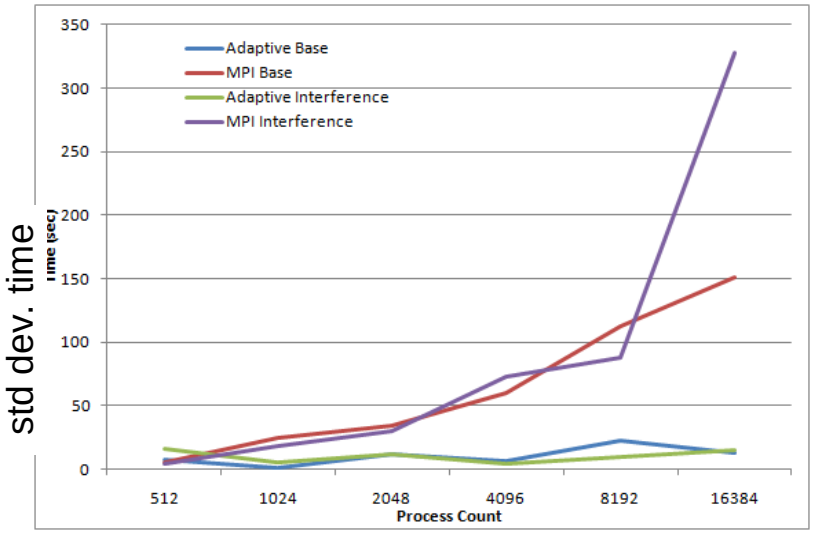
Performance Optimizations for Write

- New technologies are usually constrained by the lack of usability in extracting performance
- Next generation I/O frameworks must address this concern
  - Partitioning the task of optimizations from the actual description of the I/O



- Innovate to deal with high I/O variability, and increase the "average" I/O performance





ADIOS API

Code which writes data

```
call adios_open (adios_handle,
    "writer2D", filename, "w",
    group_comm, adios_err)
#include "gwrite_writer2D.fh"
call adios_close (adios_handle,
    adios_err)
```

XML for mapping C/F90 variables

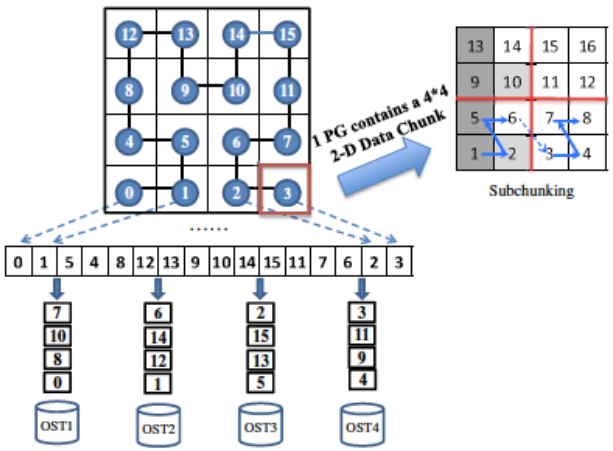
```
<adios-group name="writer2D" >
 <global-bounds dimensions="gdx,gdy"
      offsets="ox,oy">
   <var name="xy" type="real"
        dimensions="ldx,ldy"/>
</global-bounds>
</adios-group>
<transport group="writer2D"
      method = "MPI" />
```

Code which reads data

```
call adios_set_read_method (BP   ,ierr)
call adios_read_init (group_comm, ierr)
call adios_fopen (fh, fn, group_comm, gcnt, adios_err)
call adios_gopen (fh, gh, "writer2D", vcnt, acnt, adios_err)
call adios_read_var (gh, "gdx", offset, readsize, gdx, read_bytes)
call adios_read_var (gh, "gdy", offset, readsize, gdy, read_bytes)
! … calculate offsets and sizes of xy to read in…
call adios_read_var (gh, "xy", offset, readsize, xy, read_bytes)
call adios_gclose (gh, adios_err)
call adios_fclose (fh, adios_err)
```

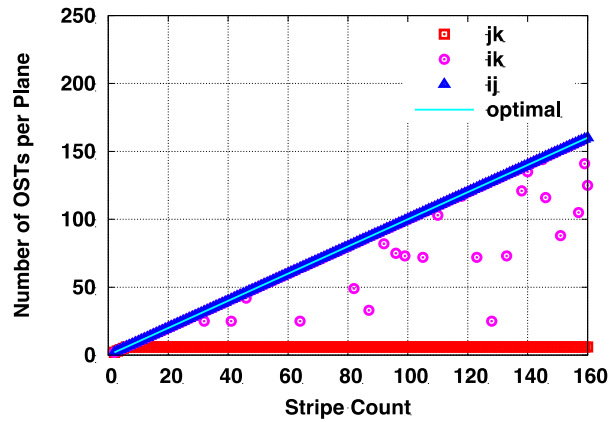
Understand why: (Examine reading a 2D plane from a 3D dataset)

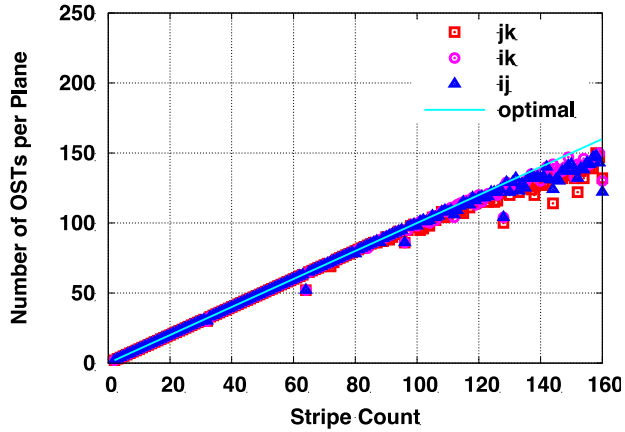
- Use Hilbert curve to place chunks on lustre file system with an Elastic Data Organization



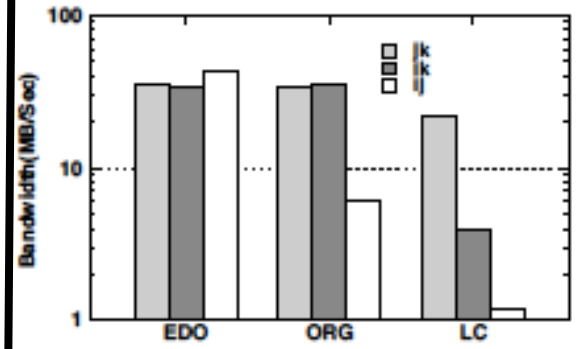
(b) SFC-based Placement with EDO

Theoretical concurrency

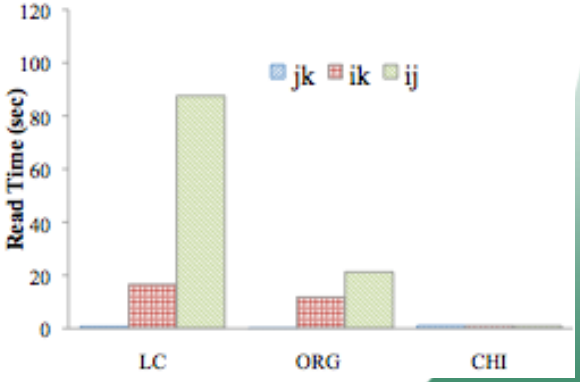




Observed Performance

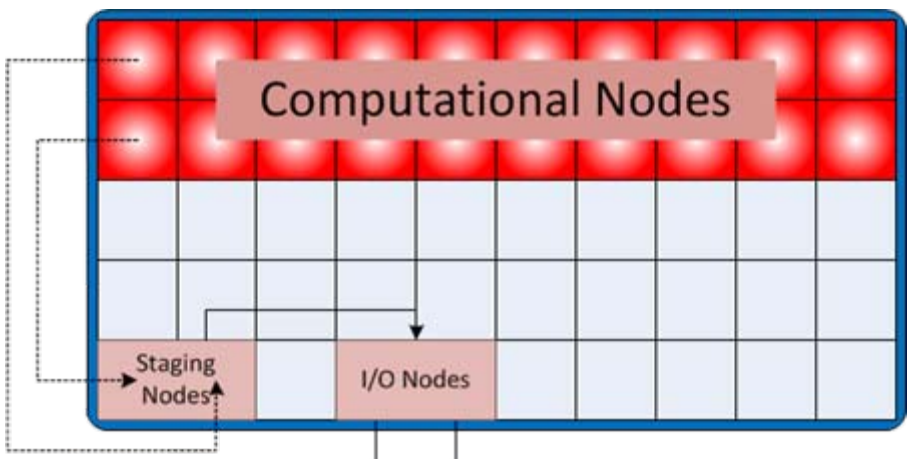


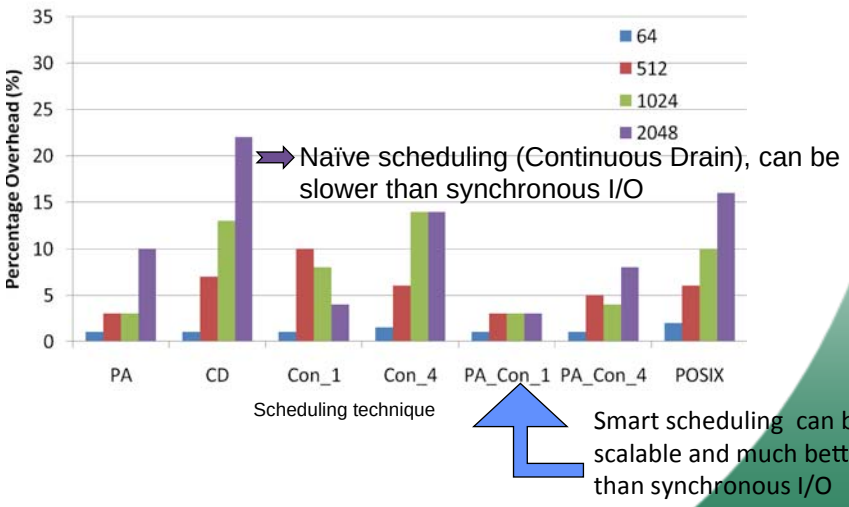
(a) Case S (stripe=128)



Designing ADIOS 2.0

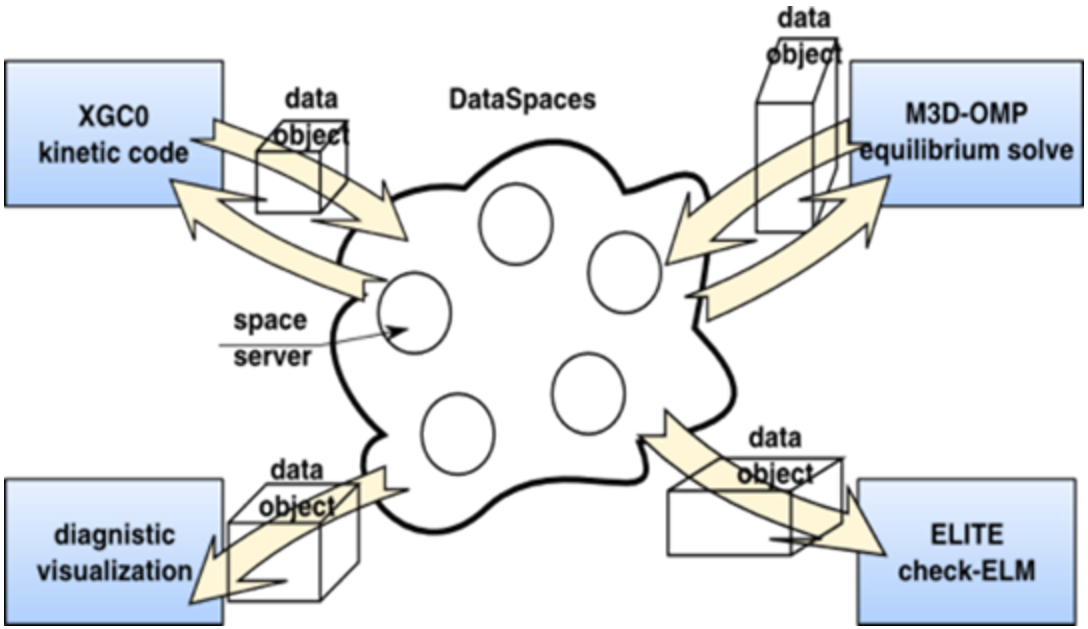
- Why staging?
  - Decouple file system performance variations and limitations from application run time
  - Allows services to run as separate executables on independent nodes from an application
- Enables optimizations based on dynamic number of writers
- High bandwidth data extraction from application: RDMA
- Scalable data movement with shared resources requires us to manage the transfers





Approach allows for in-memory code coupling

- Semantically-specialized virtual shared space
- Constructed on-the-fly on the cloud of staging nodes
  - Indexes data for quick access and retrieval
- Complements existing interaction/coordination mechanisms
  - In-memory code coupling becomes part of the I/O pipeline



Change to staging:

Code which writes data

```
call adios_open (adios_handle,
    "writer2D", filename, "w",
    group_comm, adios_err)
#include "gwrite_writer2D.fh"
call adios_close (adios_handle,
    adios_err)
```

XML for mapping C/F90 variables

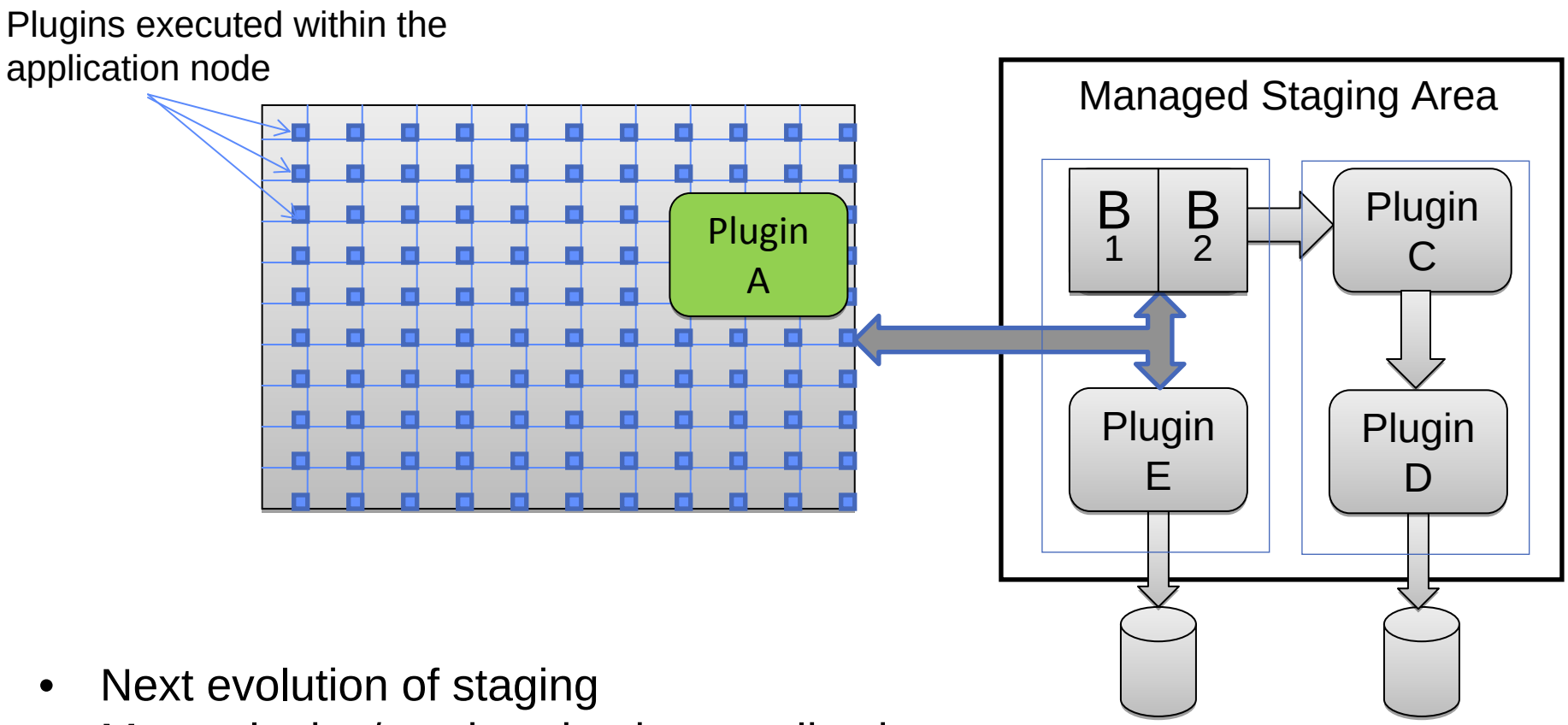
```
<adios-group name="writer2D" >
 <global-bounds dimensions="gdx,gdy"
      offsets="ox,oy">
   <var name="xy" type="real"
        dimensions="ldx,ldy"/>
</global-bounds>
</adios-group>
<transport group="writer2D"
      method = "DART"/>
```

Code which reads data

```
call adios_set_read_method (DART ,ierr)
call adios_read_init (group_comm, ierr)
call adios_fopen (fh, fn, group_comm, gcnt, adios_err)
call adios_gopen (fh, gh, "writer2D", vcnt, acnt, adios_err)
call adios_read_var (gh, "gdx", offset, readsize, gdx, read_bytes)
call adios_read_var (gh, "gdy", offset, readsize, gdy, read_bytes)
! … calculate offsets and sizes of xy to read in…
call adios_read_var (gh, "xy", offset, readsize, xy, read_bytes)
call adios_gclose (gh, adios_err)
call adios_fclose (fh, adios_err)
```

Now we have memory to memory coupling

Hybrid staging



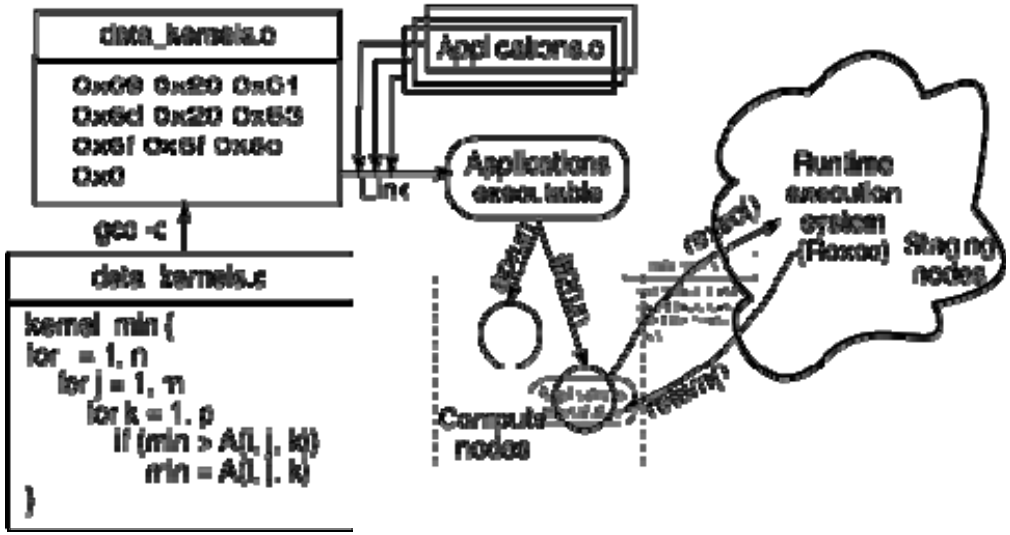
- Next evolution of staging
- Move plugins/services back to application
- Largest computational capacity
- Integrate with plugins in the remote staging area
- Need runtime system to schedule service, and need a programming model that simplifies the building and execution of the services

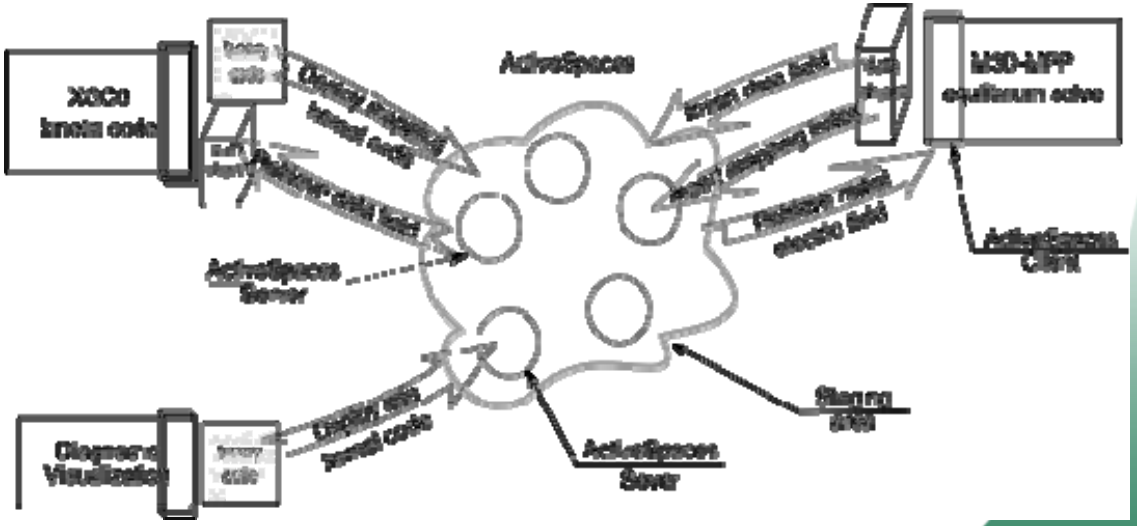
But we need to still reduce the amount of data

- We can use both lossy and lossless compression methods to reduce the data
  - ISABELA, ISOBAR, ….
- Can reduce the amount of data by 10X (lossy) for some datasets, with set accuracy
- But still need to index to query: Reduce the amount data being sent over the network
  - FastBit, ISABELA-QA, …
- We can also generate a multi-resolution stream-based compression method
  - Good way to allow users to choose ROI and zoom into higher resolution versions when necessary

Move work to data (II/II)

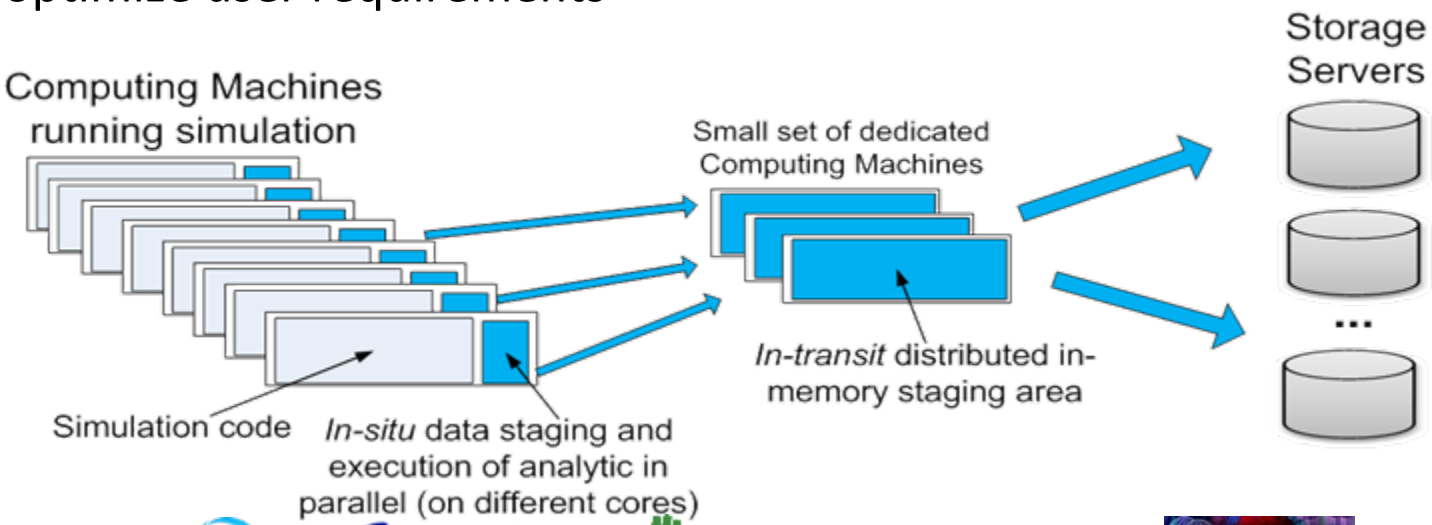
- *ActiveSpaces* – Dynamic ***binary code*** deployment and execution
- Programming support for defining custom data kernels using native programming language
  - Operates only on data of interest
- Runtime system for binary code transfer and execution in the staging area





What is ADIOS 2.0

- ADIOS 2.0 uses a SOA to deal with the complexities of diverse users, developers, and applications
- Using the ADIOS middleware to
  - Provide a service oriented architecture for easy composition of pipelines
  - Create an environment where applications can abstract data movement/IO as read/writes/queries
  - Analysis and visualization services can be plugged into the framework
  - Self describing data allows framework to move components in the pipeline to optimize user requirements



ADIOS 2.0 Pipeline Co-Design

Metrics to evaluate the design

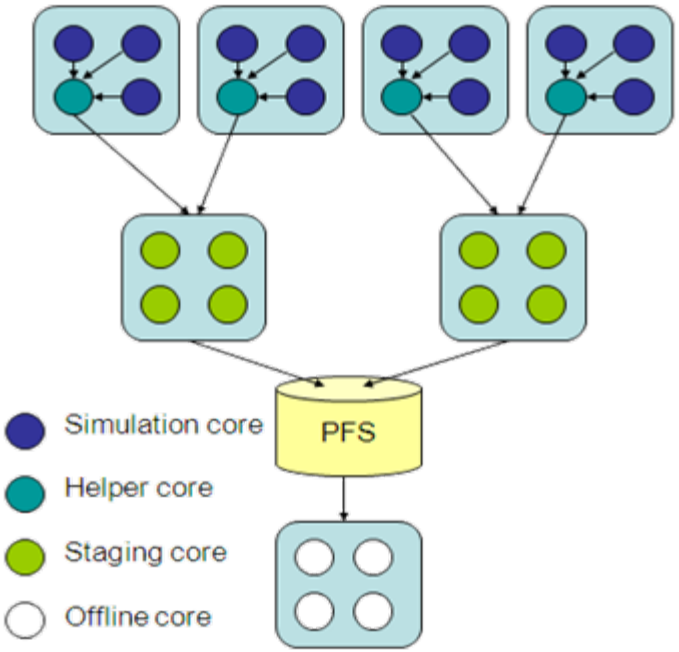
-*Performance*: **Total Execution Time** of both simulation and analytics

-*Cost*: **CPU hours** charged for simulation and analytics

-*Time to Data*: **Delay** between data generation and analytics results

-*Data movement* ~ **Power** consumption

-*Programmability*: **Usability** will greatly drive both adoption and optimization

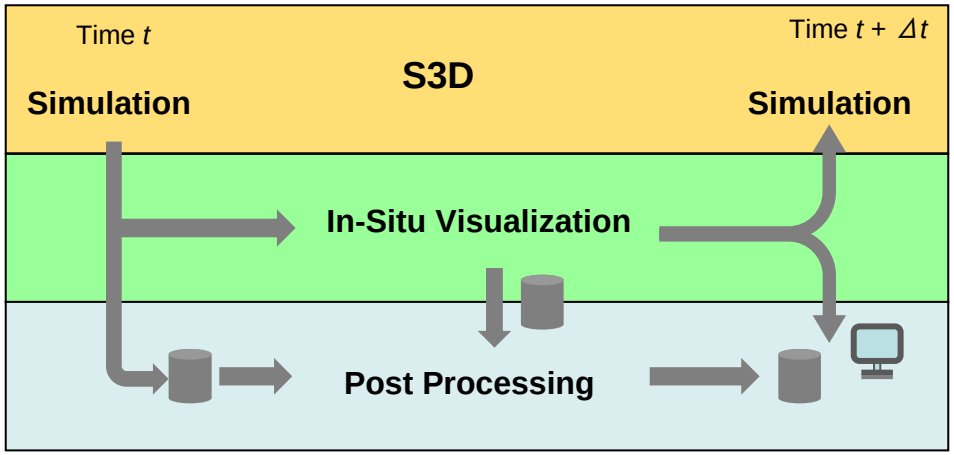


Design Goals

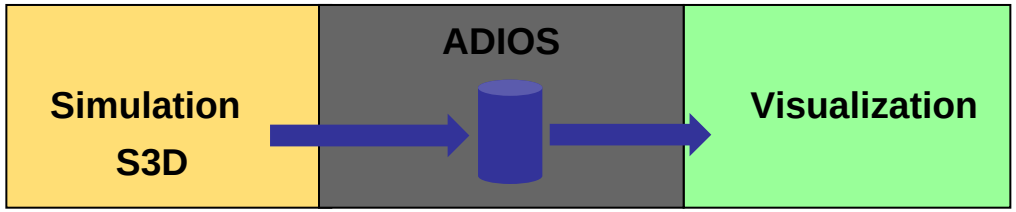
- *Flexible Placement:* Data movement can be reduced through intelligent placement
- *Hybrid Staging:* Exploit deep memory through hybrid staging which can use on-chip and on-node resources to optimize the data pipeline
- *Performance*: Optimized write and read I/O throughput in addition to the value added through the data analysis and visualization pipeline

Insitu Visualization through SDMA middleware

Directly integrated with simulation



Connected through ADIOS



- Accomplishments
  - Used ADIOS as the "connector" between visualization and simulation
  - Visualization can not execute insitu using the same resources as the application
  - It can also be executed intransit using discrete resources close to the application
- Impact
  - Provides an example design of a data processing service
  - Allows co-design with data management to develop type system for combustion
  - Allows co-design with data management runtime to develop support for flexible placement and data transport optimizations

Conclusions

- ADIOS uses a SOA approach to
  - Get semantic knowledge of the data for code coupling, in-transit visualization, analysis
  - Teams must work together to bridge the GAP between research and production
- Hybrid staging and in situ workflow execution will allow users to intertwine applications, libraries, middleware for complex analytics
- Collaboration is critical to the success of this field
- Now looking for state-of-the-art analytics and visualization routines to be plug-into our system