Barbara Chapman
Stony Brook University
Brookhaven National Laboratory

BROOKHAVEN NATIONAL LABORATORY

a passion for discovery

U.S. DEPARTMENT OF ENERGY | Office of Science

How To Get Tied Up In Knots

Barbara Chapman
Stony Brook University
Brookhaven National Laboratory

BROOKHAVEN NATIONAL LABORATORY
a passion for discovery

U.S. DEPARTMENT OF ENERGY | Office of Science

(Near) Real-Time Big Data Streaming Analysis

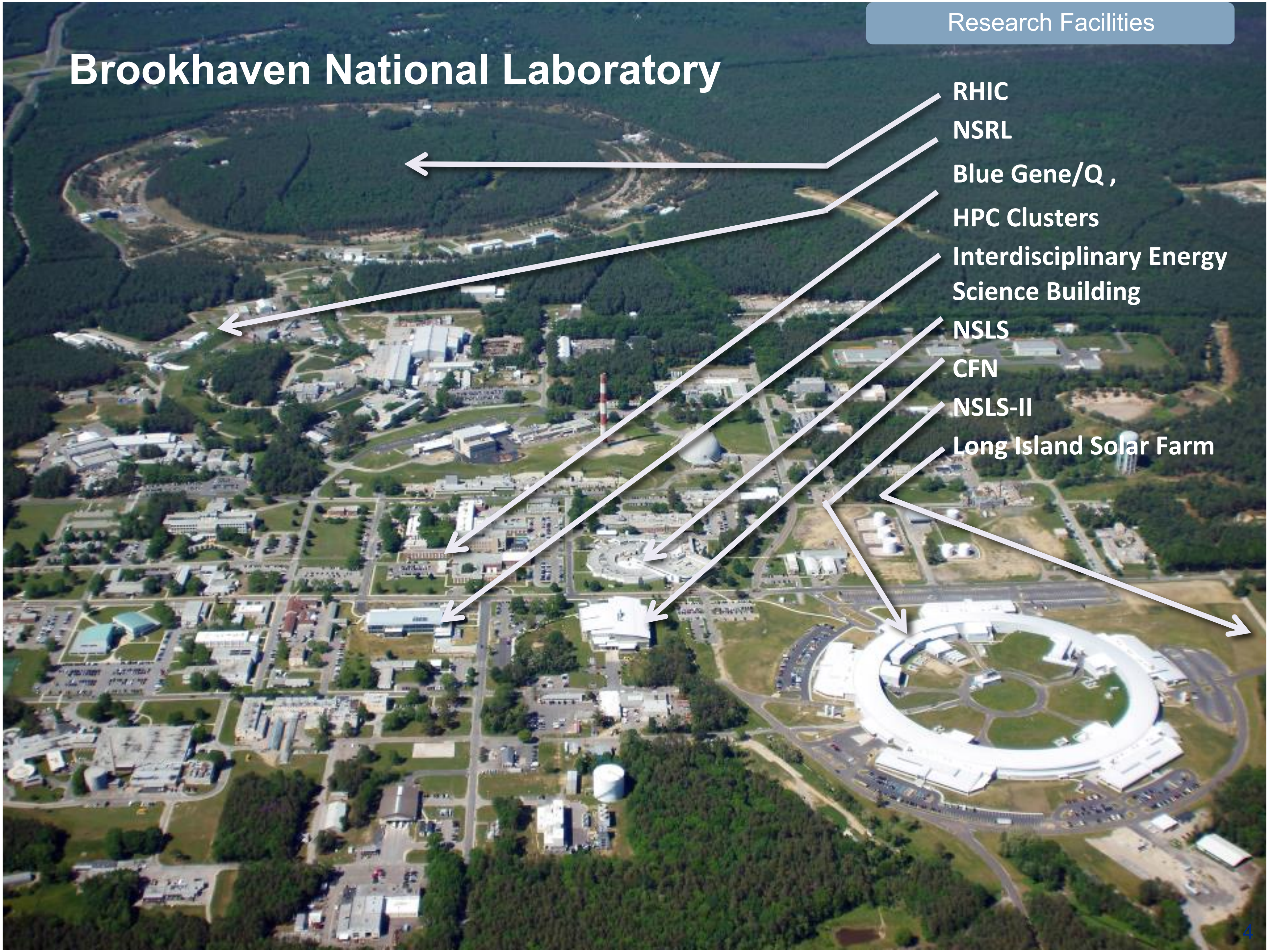
Barbara Chapman
Stony Brook University
Brookhaven National Laboratory

BROOKHAVEN NATIONAL LABORATORY

a passion for discovery

U.S. DEPARTMENT OF ENERGY | Office of Science

Brookhaven National Laboratory



RHIC

NSRL

Blue Gene/Q ,

HPC Clusters

Interdisciplinary Energy Science Building

NSLS

CFN

NSLS-II

Long Island Solar Farm

Major Research Facilities

RHIC

- 2.4 mile circumference
- Studying the origins of universe through ion collisions revealing make up of visible matter
- Discovery of the ‘perfect liquid’

National Synchrotron Light Source II

National Synchrotron Light Source II

- Soon to be world’s brightest X-ray light source
- $960 million project - hundreds of local jobs
- Completed in 2014
- Approx. 3,000 visiting researchers

Center for Functional Nanomaterials

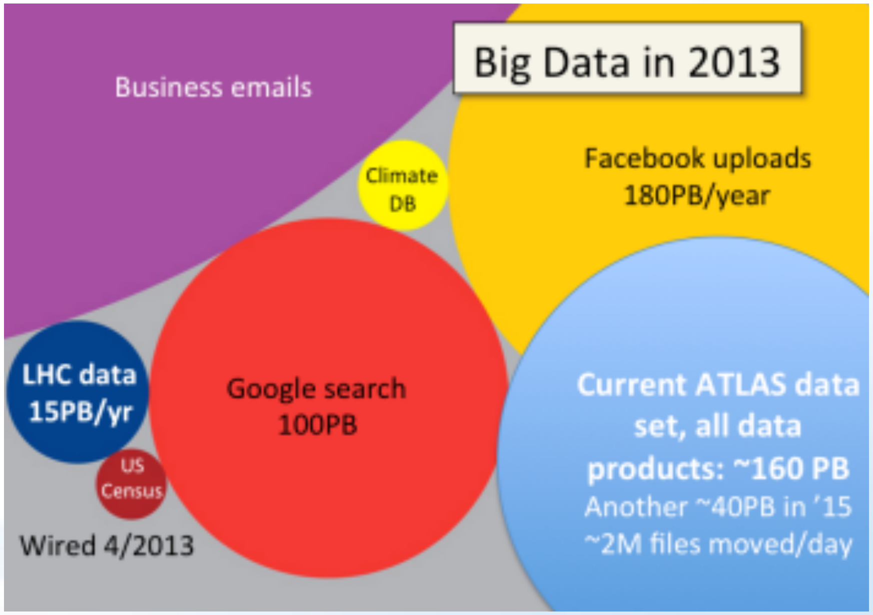
Center for Functional Nanomaterials

- Exploring energy science at the nanoscale
- Building new materials atom-by-atom to achieve desired properties and functions

Big Data Computing in HEP and NP

RHIC ATLAS Computing Facility (RACF) & Physics Applications Software (PAS) Groups, BNL Physics Dept

- RACF
  - 15 years of experience at the largest data scales
  - Data sets on order of 100PB (ATLAS is 160 PB today)
- PanDA, LHC's exascale workload manager developed at BNL
  - 2013: ~1.3 Exabytes in 200M jobs, ~150 sites, ~1000 users
  - Continuous innovation needed for scaling: ATLAS data volume increasing 10X in 10 years
  - Intelligent networks, agile workload management, distributed data handling



Next Generation Workload Management and Analysis System For Big Data: Big PanDA

PI: Alexei Klimentov; BNL PAS Group : T.Maeno, S.Panitkin, T.Wenaus; BNL CSI : D.Yu

http://pandawms.org/info

Science Objectives & Impact

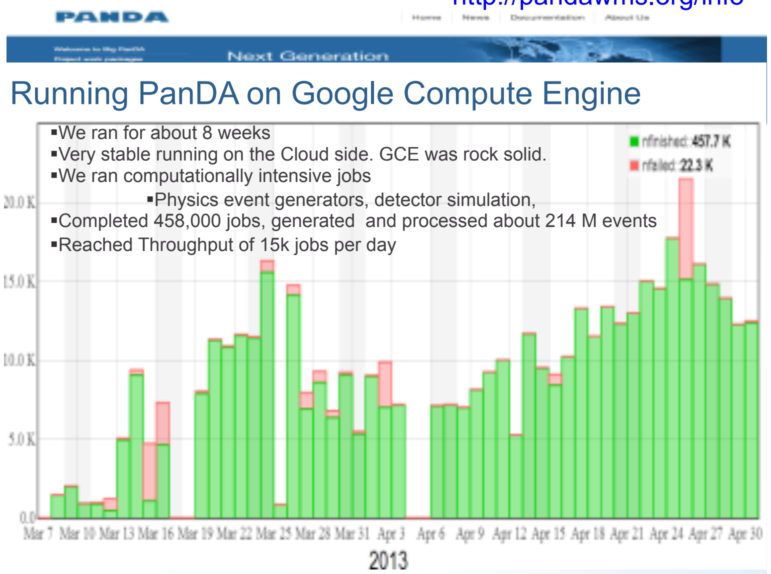
Objectives :

- Factorizing the core components of PanDA
- Evolving PanDA to support extreme scale computing clouds and Leadership Computing Facilities
- Integrating network services and real-time data access to the PanDA workflow
- Real time monitoring and visualization package for PanDA

Impact :

- Enable adoption of PanDA by a wide range of exascale scientific communities
- Provide access to a wide class of distributing computing to data intensive sciences
- Introduce the concept of Network Element as a core resource in workload management
- Provide easy to use and easy to virtualize interface for scientific communities

Multiple DOE-supported institutes: BNL, ORNL, ANL, LBNL and US Universities : UTA, Rutgers



Running PanDA on Google Compute Engine

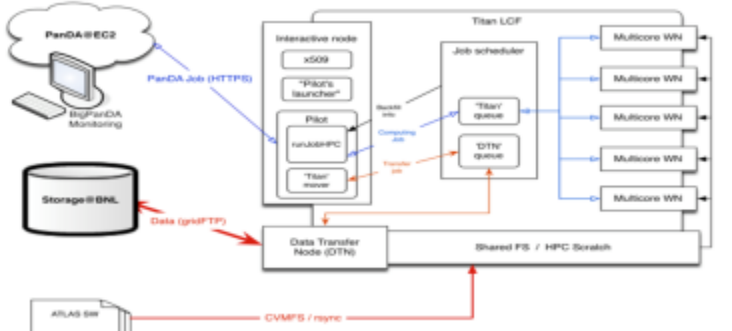
- We ran for about 8 weeks
- Very stable running on the Cloud side. GCE was rock solid.
- We ran computationally intensive jobs
  - Physics event generators, detector simulation,
- Completed 458,000 jobs, generated and processed about 214 M events
- Reached Throughput of 15k jobs per day

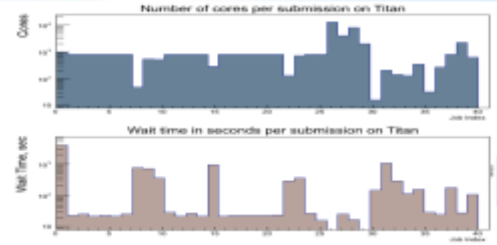
Progress & Accomplishments

- Basic PanDA code (server and pilot) is factorized
- PanDA instance at Amazon EC2 is set up (VO independent)
- Common project with Google was successfully completed
- First implementation of PanDA workflow management system on leadership supercomputer (Titan)
  - Also NERSC and Anselm (Ostrava)
- Successful access to large, otherwise-unavailable opportunistic resources.
- Successful operation of multiple applications required by high energy physics and high energy nuclear physics experiments.
- Networking throughput performance and P2P statistics collected by different sources continuously exported to PanDA database

Brookhaven Science Associates

Running PanDA on Oak Ridge LCF (Titan)





Short wait times !
Average wait ~4.2 min
Min wait ~16 sec
Max wait ~55.3 min

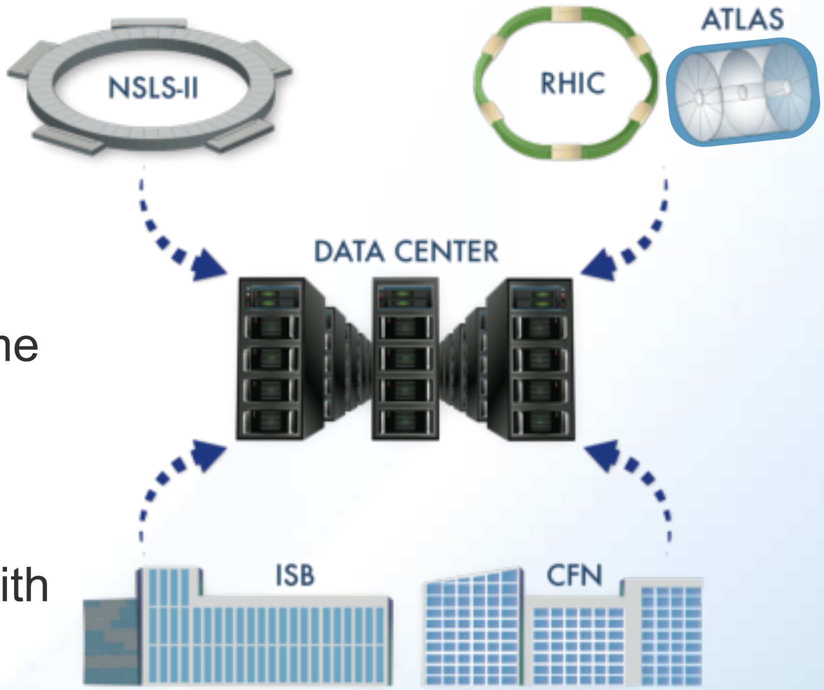
Number of cores per opportunistic Titan job and associated wait times over the course of 24 hour test.

Computational Science Initiative

Vision: Expand and leverage BNL’s leadership in the analysis and processing of large volume, heterogeneous data sets for high-impact science programs and facilities

To achieve this vision BNL has:

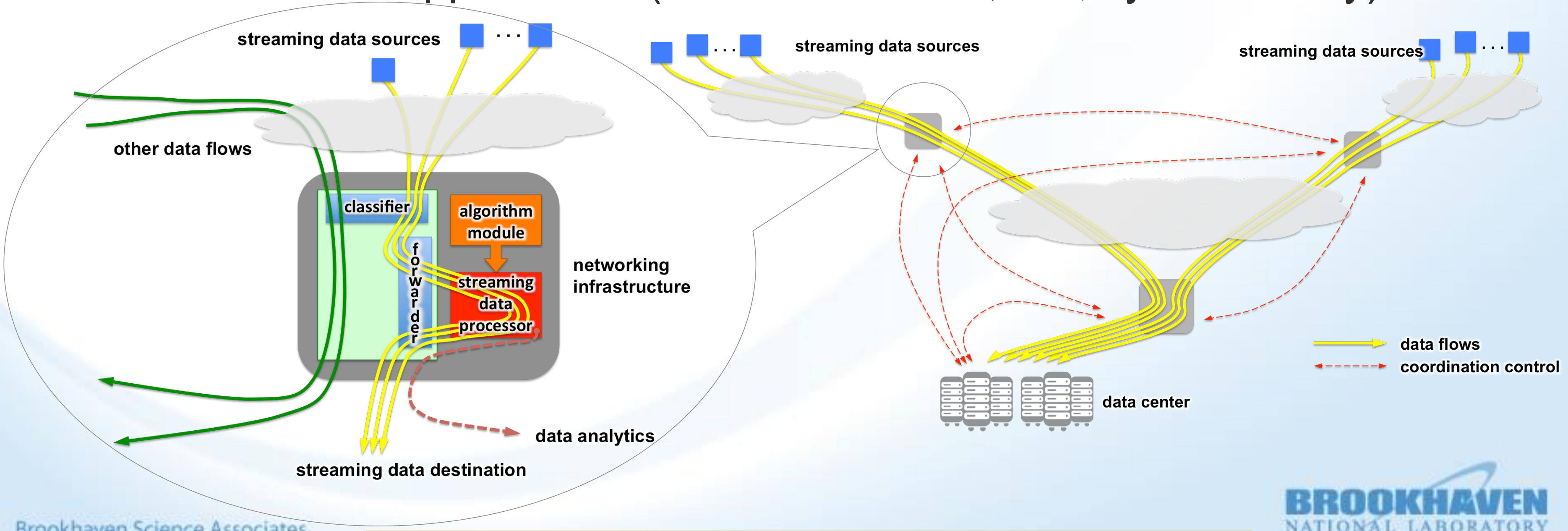
- Created Lab-level Computational Science Initiative reporting to DDST
- Begun to build Lab-wide sustainable infrastructure for data management, real-time analysis and complex analysis
  - Initial focus: NSLS-II
- Initiated growth of competencies in applied mathematics & computer science aligned with the missions of ASCR, other SC programs
- Established partnerships with SBU, key universities, IBM, Intel, other National Labs



Intelligent Networking for Streaming Data

D. Katramatos, S. Yoo, K. Kleese van Dam, CSI

- Streaming Data Analysis on the Wire (AoW)
  - Research and develop framework that enables generic computation on data on the wire, i.e. while in transit in the network
  - Primary goal: provide real-time/near real-time information to facilitate early decision making
    - Data analysis
    - Simple transformations
    - Pattern detection
  - Multitude of applications (sensor networks, IoT, cybersecurity)



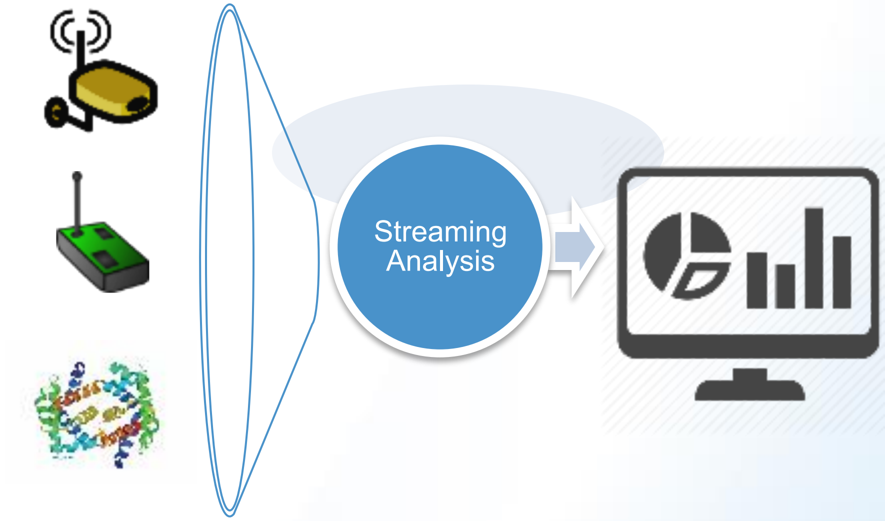
- https://www.bnl.gov/compsci/projects/analysis-on-the-wire.php

(Near-)Realtime Streaming Analytics

Shinjae Yoo (CSI), Dmitri Zakharov (CFN), Eric Stach (CFN), Sean McCorkle (Biology)

Summary and significance

- Streaming analytics is one of the most attractive approach to handle high velocity and high volume data algorithmically due to one pass and limited memory operation
- Our streaming learning algorithms showed performance comparable to batch learning algorithms and superior to legacy streaming algorithms



Data research and capabilities

- Built streaming manifold learning algorithms, which can be applicable to most of unsupervised learnings including feature selection, anomaly detection, and clustering analysis
- Develop streaming analytics algorithms, customized to handle unique challenges in streaming analytics
- Applying streaming analytics on various science problems starting from CFN

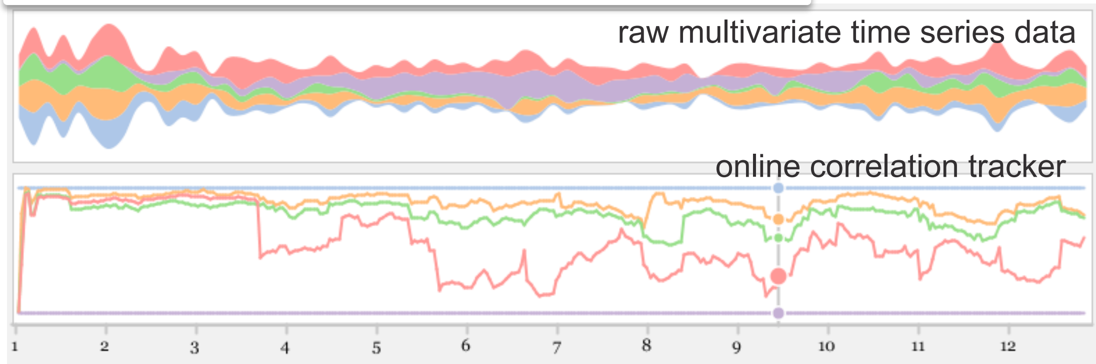
Data frontiers

- CFN: near real time analysis of transmission electron microscopy (TEM) images from a 3GB/s image stream
- Biology: processing all known protein pairs to get new level of biological insights
- NSLS-II: applicable to high velocity beamlines at NSLS-II.
- SmartGrid: distributed high velocity data such as PMU for distributed state estimation

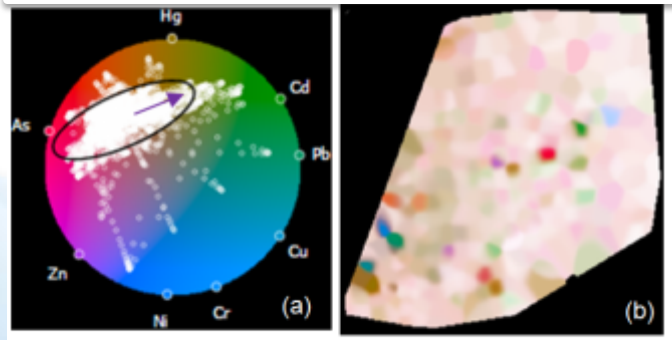
Streaming Visual Analytics and Visualization
W. Xu, Computational Science Initiative

- Enable visual data interaction including browsing, comparison, and evaluation to steer streaming data acquisition and online data analysis.

Streaming data correlation analysis



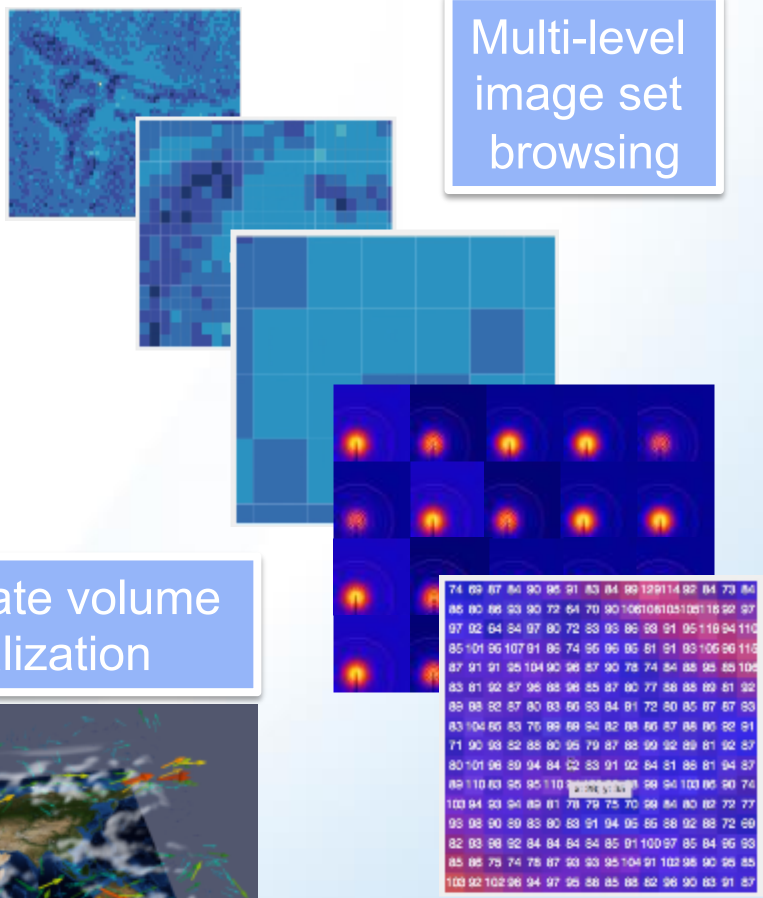
Correlation-driven color mapping



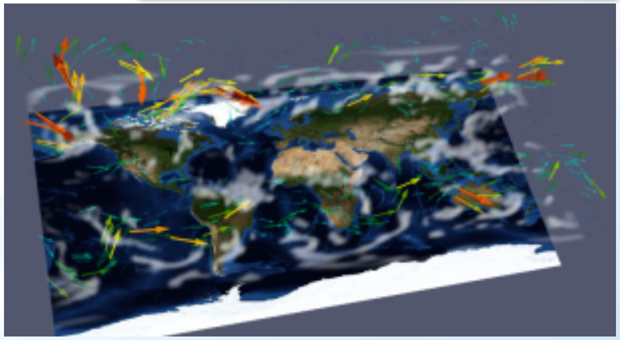
HCL color palette

Air pollutants distribution over certain region

Multi-level image set browsing



Multivariate volume visualization



CREDIT: CoE for Big Military Data Intelligence

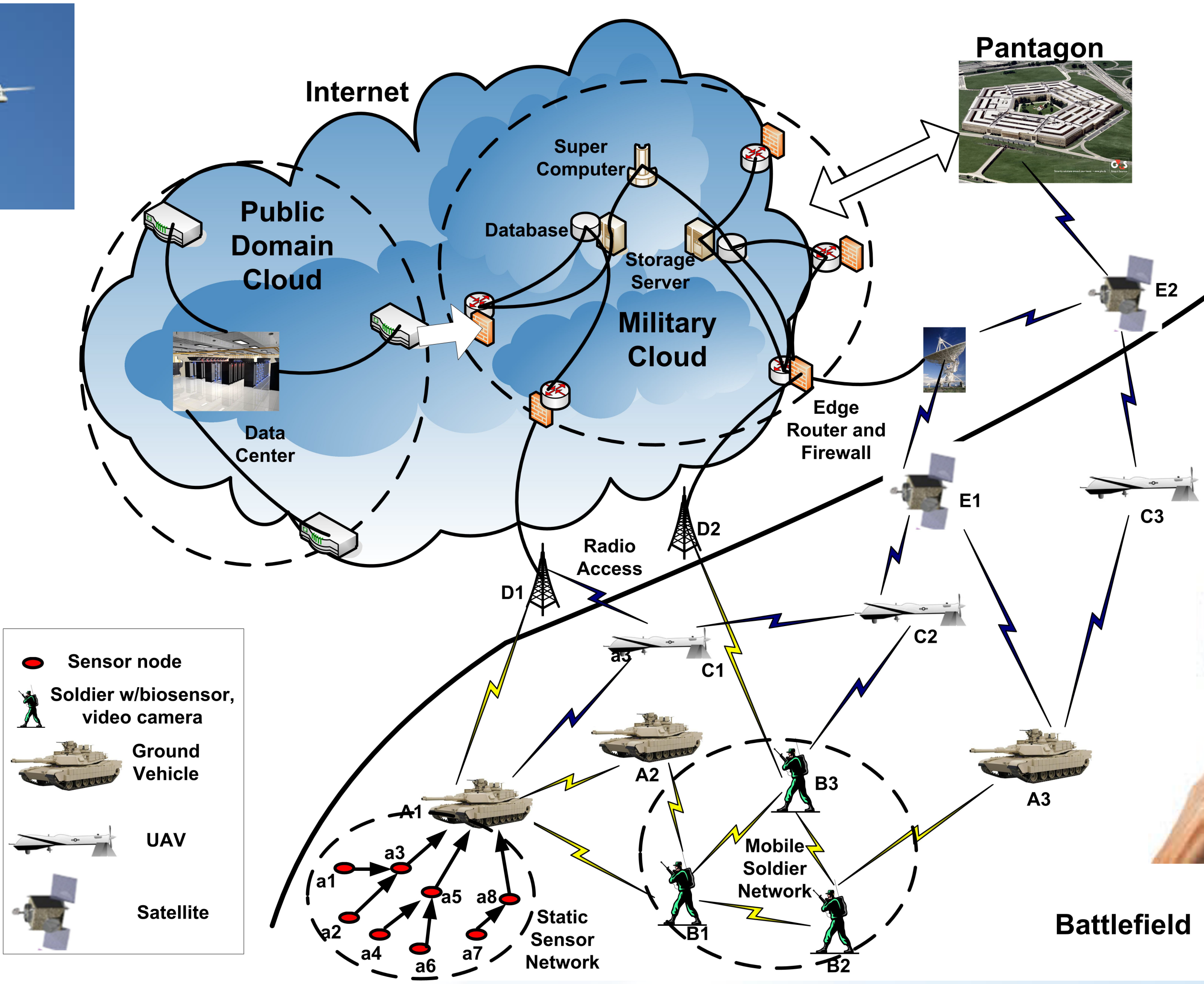
- Big-data real-time analytics research
  - Sophisticated battlefield data fusion and analytics
  - Integrated, scalable data analysis and inference infrastructure
- Multiple sources of data, some real-time, potentially unreliable
  - High volume, velocity, variety; variable, uncertain quality (veracity)
- Stringent requirement for real-time decision-making

- Novel machine-learning algorithms for high-dimensional heterogeneous data sets with missing data
  - Deep learning for advanced feature detection
  - Critical event detection
- Enhancements to Spark for battlefield data, scheduling with real-time constraints, optimization for accelerator-based architectures
- Visualization on large screen and mobile devices

- Collaborators: Prairie View A&M, Stony Brook

Stony Brook University

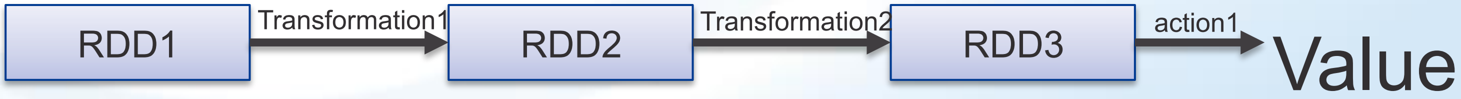
CREDIT Real-Time Detection and Decision-Making



Stony Brook University

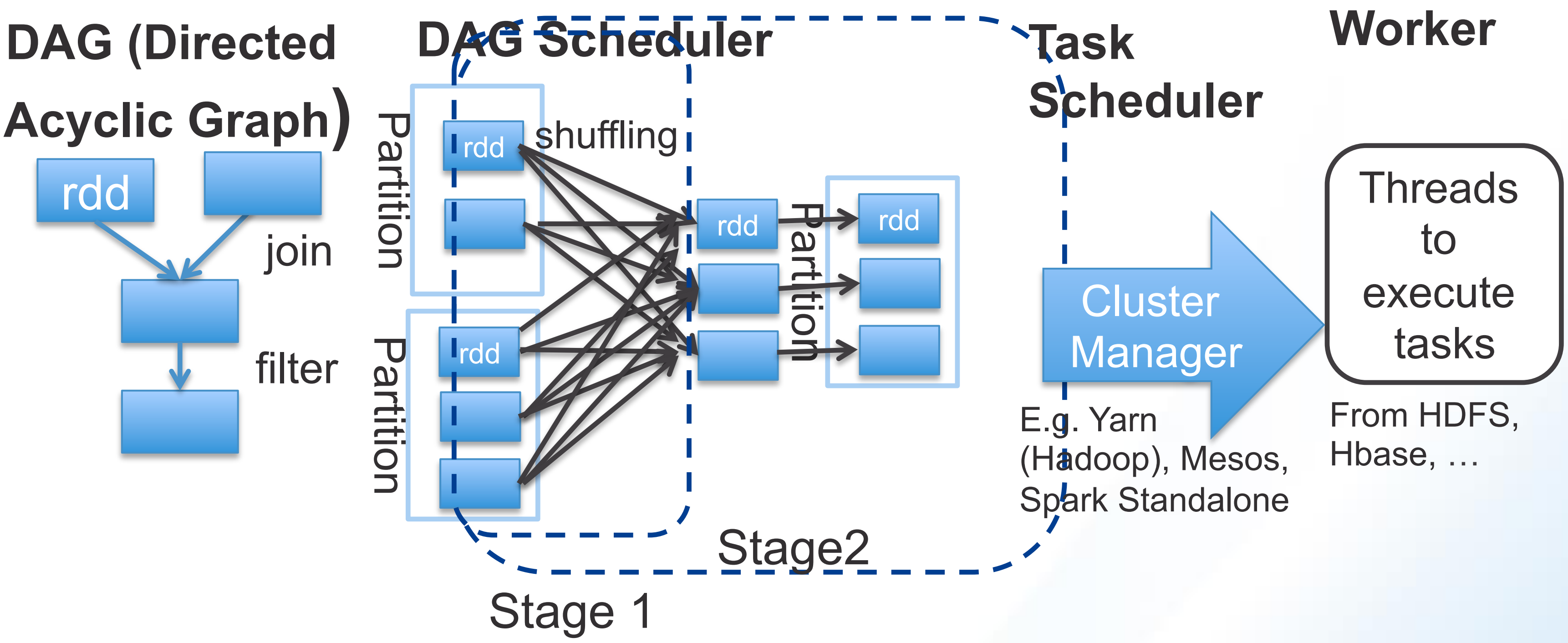
Spark: Resilient Distributed Data (RDD)

- Core data management concept in Spark
- Read-only datasets
- Each RDD transforms to another RDD (map, filter, etc)
- Lazy evaluation: RDD values do not materialize unless an **action** is required (count, collect, save, etc)
- Fault-tolerance is managed using lineage of the RDDs
- A dataset is (resiliently) distributed across the cluster nodes: no single node has all the data, possible recovery from node failures
- In-memory processing: storing computed data across jobs for reuse
- Application Domain: iterative machine learning algorithms and interactive data mining tools



RDD1 Transformation1 RDD2 Transformation2 RDD3 action1 Value

Spark vs. MPI Execution Model



DAG (Directed Acyclic Graph)

rdd

join

filter

DAG Scheduler

Partition

rdd

shuffling

rdd

Partition

rdd

rdd

Partition

Stage 1

Stage2

Task Scheduler

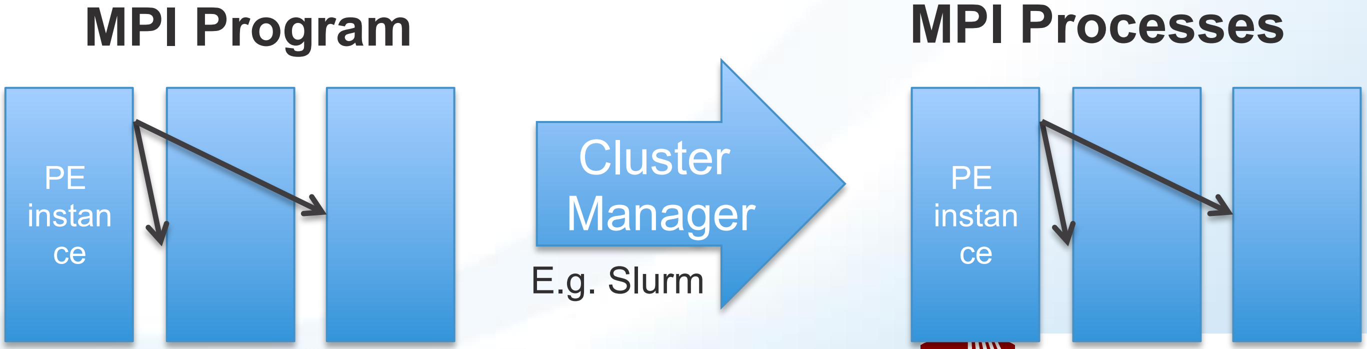
Cluster Manager

E.g. Yarn (Hadoop), Mesos, Spark Standalone

Worker

Threads to execute tasks

From HDFS, Hbase, …



MPI Program

PE instance

Cluster Manager

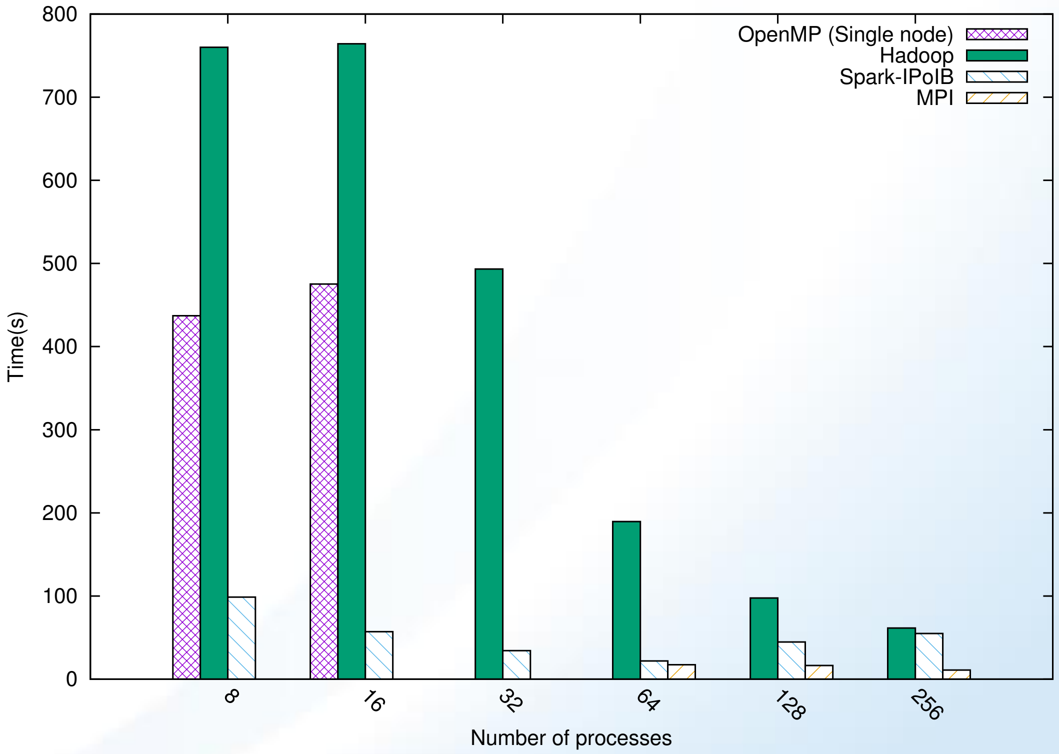
E.g. Slurm

MPI Processes

PE instance

StackExchange *AnswersCount* Benchmark

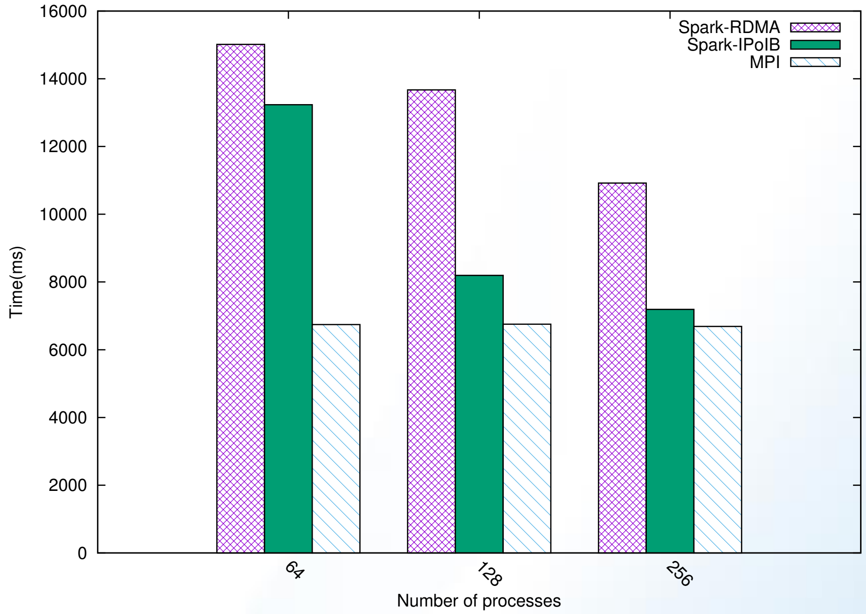
- Counts average number of answers to a query
- 80GB test data set
- Hadoop saves intermediate data to disk; Spark minimizes disk use
- OpenMP unoptimized
- MPI: could not handle very large files
- Spark scales well up to 64 processes



https://github.com/hrasadi/HPCfBD

BigDataBench PageRank

- BigDataBench implementation of PageRank in Scala
- 16 processes/node, 1,000,000 vertices on SDSC COMET
- Spark with data caching scales well
- Spark's RDMA does not help since little data motion

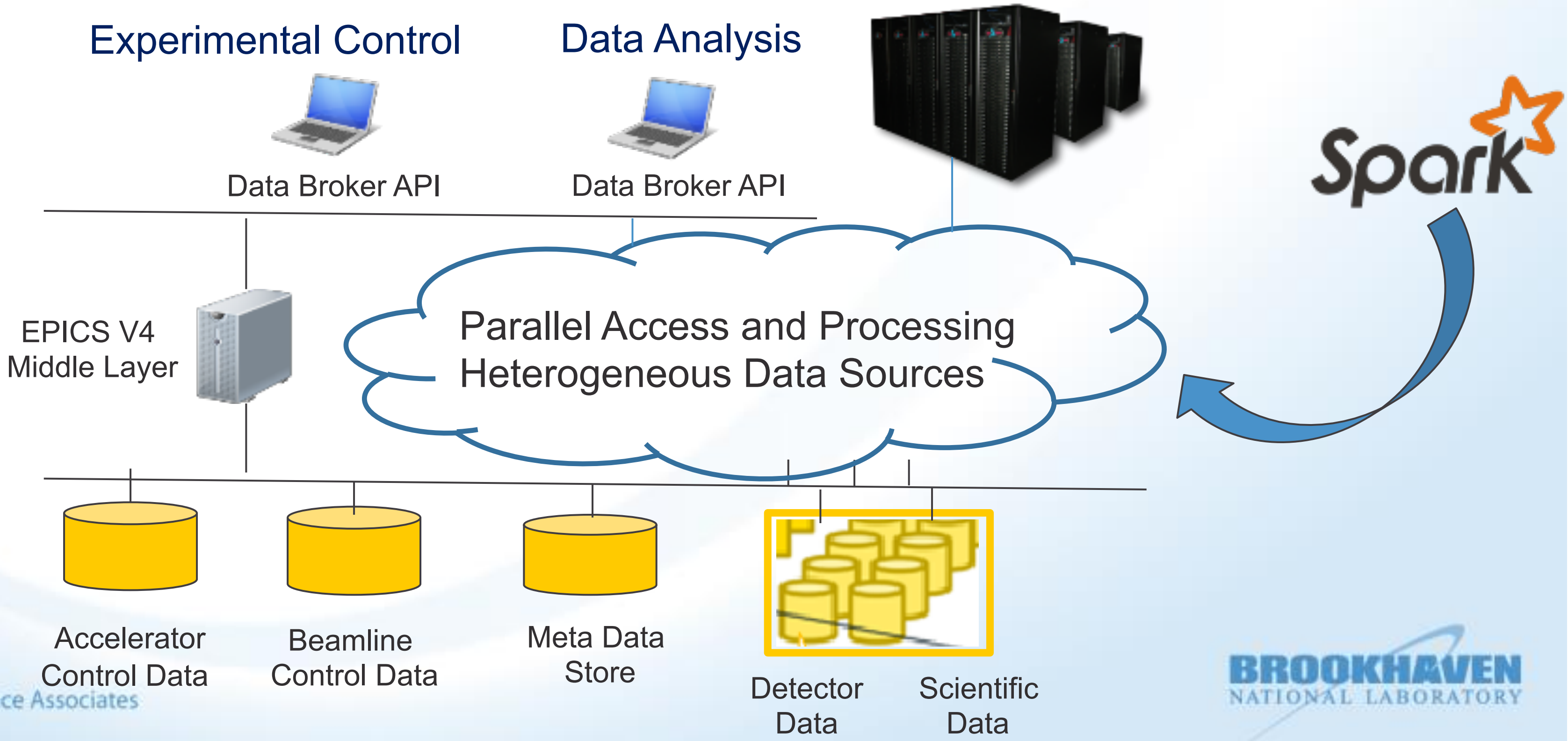


```
var ranks = links.mapValues(v => 1.0).persist(StorageLevel.MEMORY_AND_DISK)
for (i <- 1 to iters) {
  val contribs = links.join(ranks).values.flatMap {
    case (urls, rank) => val size = urls.size
    urls.map(url => (url, rank / size))
  }.persist(StorageLevel.MEMORY_AND_DISK)
  // This caching is not done in HiBench Implementation
  ranks = contribs.reduceByKey(_ + _).mapValues(0.15 + 0.85 * _)
}
```

Integrated Platform for Data-Intensive Science

N. Malitsky, NSLS II Control Department, BNL

- Development of a generic data integration platform based on Spark
  - Managing, analyzing, and parallel processing of **heterogeneous data sources** from experimental facilities and scientific applications
  - Support for hybrid data layer combines NoSQL metadata catalogs and repositories of heterogeneous data files
  - Additional support for multi-dimensional (time-series) datasets and GPU-based image processing, etc.



TensorFlow

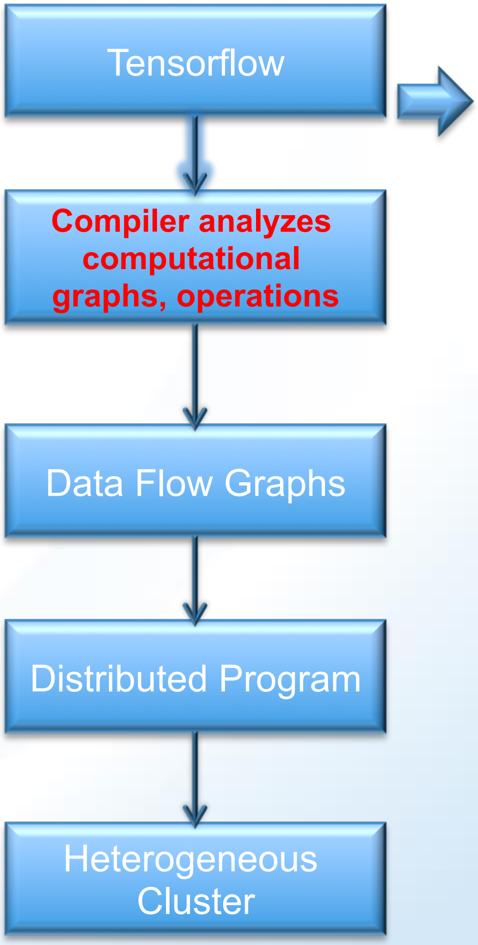
- Google’s TensorFlow: open source software, since November 2015
- C++, Python ; core of TensorfFlow written in C++
- Library of operations that manipulate *tensors* and *persistent variables*
  - Tensors are arbitrary dimensionality arrays
  - Element type may be specified or inferred at graph construction time.
  - Elementwise math operations, matrix operations, checkpointing, locks, control flow, neural net building; ML ops (stochastic gradient descent)
  - Control operations include means to express loops

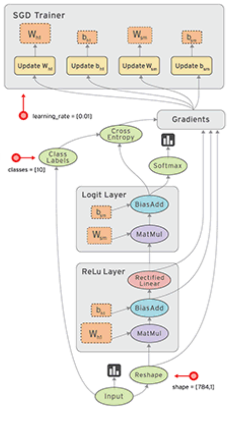
- Run operation specifies what needs to be computed (output)
- Implementation constructs execution graph of operations
  - computes transitive closure of nodes that must be executed to derive outputs
  - determines execution order that respects their dependencies
- Assumes user sets up graph once and executes it thousands or millions of times via Run calls.

Improving TensorFlow Scalability

- TensorFlow intended for parallel execution
  - Modeling phase selects resources
  - Send/receive constructs inserted
  - Better starting point for exploiting HPC systems
  - FT in messaging and periodic checks
  - Persistent variables periodically saved

- Extend interface for new algorithms
  - BNL and CREDIT partners
- Map computations in Tensorflow graph to (Data Flow) Task Graph for efficient cluster implementation
  - Instantiation of operations
- Optimize for HPC systems


Tensorflow
Compiler analyzes computational graphs, operations
Data Flow Graphs
Distributed Program
Heterogeneous Cluster


SGD Trainer
W_hl
b_hl
W_sm
b_sm
Update W_hl
Update b_hl
Update W_sm
Update b_sm
learning_rate = [0.01]
Gradients
Cross Entropy
Class Labels
classes = [10]
Softmax
Logit Layer
b_sm
BiasAdd
W_sm
MatMul
ReLu Layer
Rectified Linear
b_hl
BiasAdd
W_hl
MatMul
Reshape
shape = [784,1]
Input