Bringing Large-Scale Analytics to Accelerators

Nikolay Malitsky

11th International Computational Accelerator Physics Conference

Rostock, Germany

August 19-24, 2012

BROOKHAVEN NATIONAL LABORATORY

a passion for discovery

U.S. DEPARTMENT OF ENERGY | Office of Science

Outline

- Background
- Google Approach
- SciDB Array-Oriented Model
- GraphLab Distributed Framework on Graph
- Integrated System

NSLS-II, BNL

General parameters: 3 GeV, 500 mA; circumference: 791.5 m; lattice: 30 cell, DBA
Ultra-low emittance: ε_x, ε_y = 0.6, 0.008 nm-rad; beam size: σ_y = 2.6 μm, σ_x = 28 μm
Insertion devices: in-vacuum undulators, elliptically polarizing undulators, damping wigglers, *etc*.
Experimental techniques: tomography, diffraction microscopy, coherent diffraction imaging, *etc.*

EPICS: Experimental Physics and Industrial Control System

Collaboration :

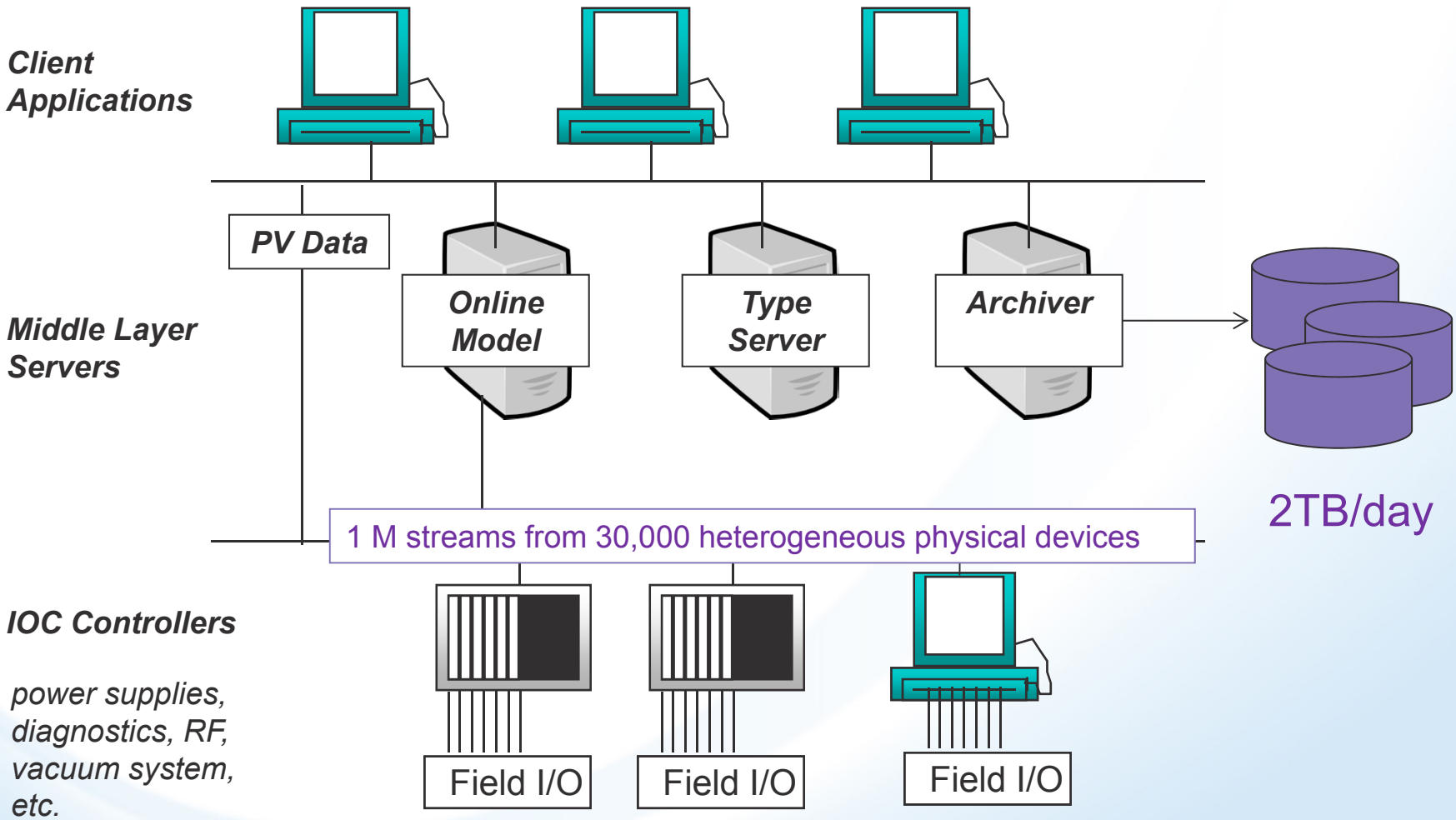
- Over 150 independent projects in North America, Europe, Africa, Australia, South America, and Asia
- Applications in particle physics, astronomy, and industrial control
- Independent development, co-development and incremental development of code done by members
- Large collaboration meetings to report new work, discuss future directions, explore new applications, and explore new requirements for existing codes

Distributed Architecture :

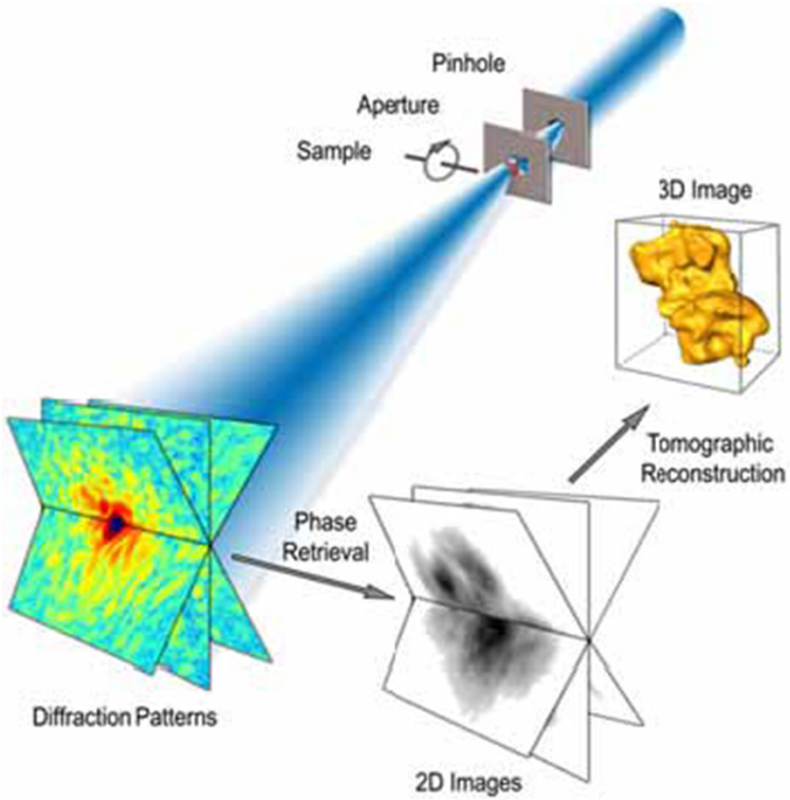
- Flat architecture of front-end controllers and operator workstations that communicate via TCP/IP and UDP
- Client/server based with independent data stores providing read/write access directly between any two points

Collection of Numerous Tools

Use Case I: Archiving and Processing Time Series of the Accelerator Control Data



Use Case II: DAQ and Analysis of Experimental Data

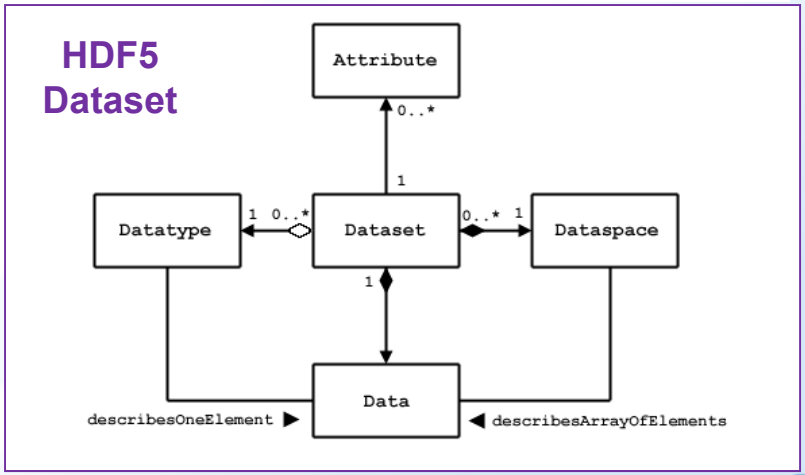


Courtesy of Qun Shen (NSLS-II)

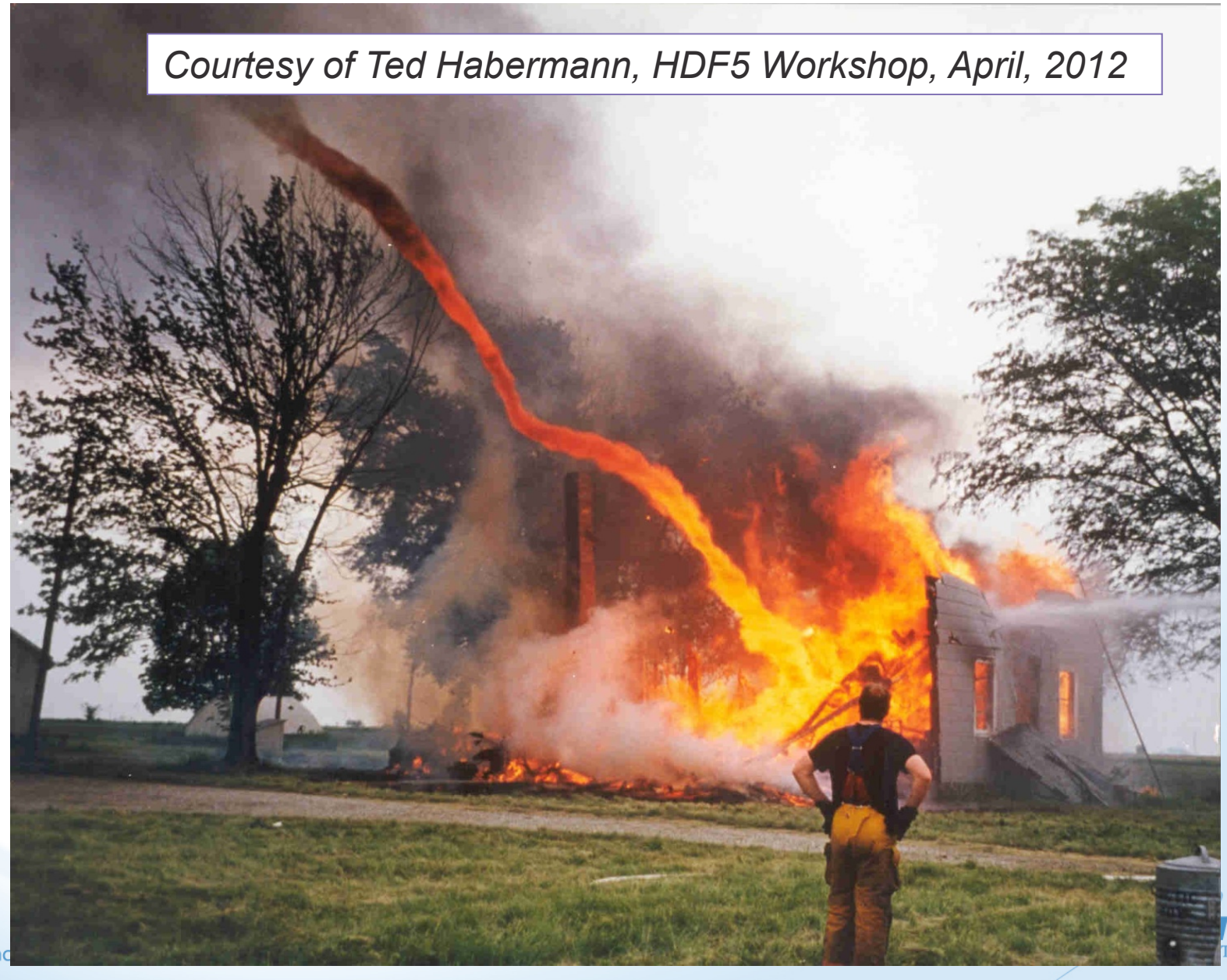
Detector data rate: ~ 50 GB /s
Storage data rate: ~ 2 TB /day

The HDF Group

HDF5 Dataset



To Summarize …



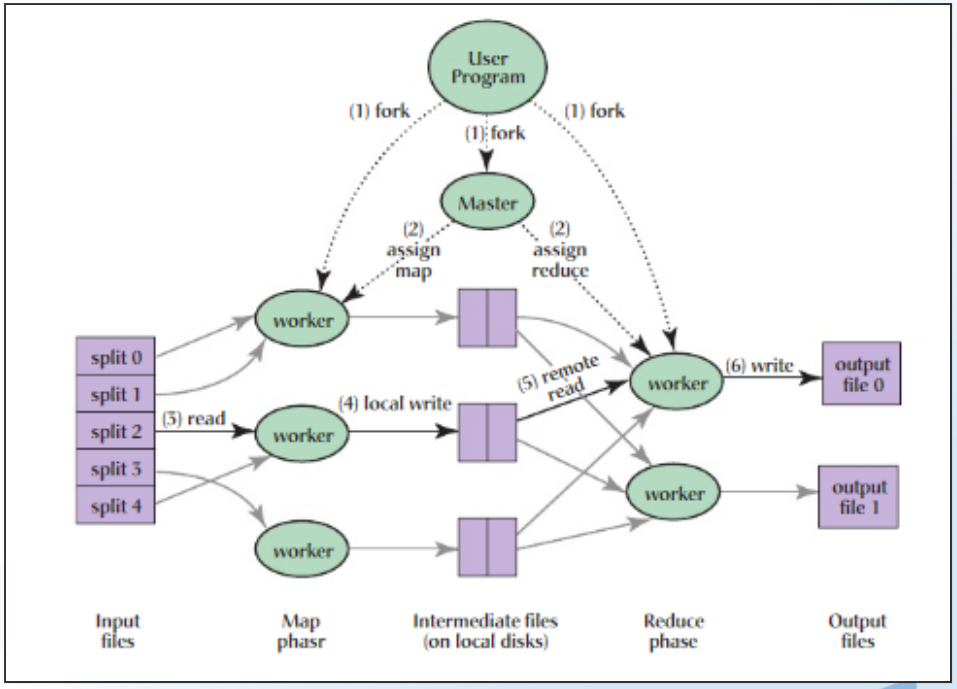
Google Approach

	Google	Apache Hadoop
Applications	Google Analytics, Google Earth, *etc.*	Hive, Mahout, etc.
Processing	MapReduce	Hadoop MapReduce
Data Model	BigTable	Hbase, Cassandra
File System	Google File System (GFS)	Hadoop Distributed File System(HDFS)

BigTable[1] is a sparse, distributed, persistent sorted map, indexed by row key, column key, and timestamp.

MapReduce[2] **Programming Model:**

- **map** (k1, v1) -> list (k2, v2)
- **reduce** (k2, list(v2)) -> v3



---

[1] **F.Chang et al.**, OSDI, 2006

[2] **J. Dean and S. Ghemawat** , OSDI 2004

But …

does not address the scientific applications relying on:

- **multi-dimensional array-oriented data model. Alternative solutions:**
  - SciDB: native array-oriented database
  - SciHadoop: Hadoop plugin for NetCDF data sets
  - RasDaMan: array execution on top of blobs in relational databases
  - MonetDB: array simulation on top of relational databases
  - ArrayStore, Pyramid, *etc.*
- **complex iterative algorithms. Alternative approaches:**
  - SciDB, Vertica, etc.: parallel databases
  - GraphLab: distributed framework on graph
  - Twister, HaLoop: MapReduce iterative extensions
  - Message-Passing Interface

SciDB: Open Source Data Management and Analytics System

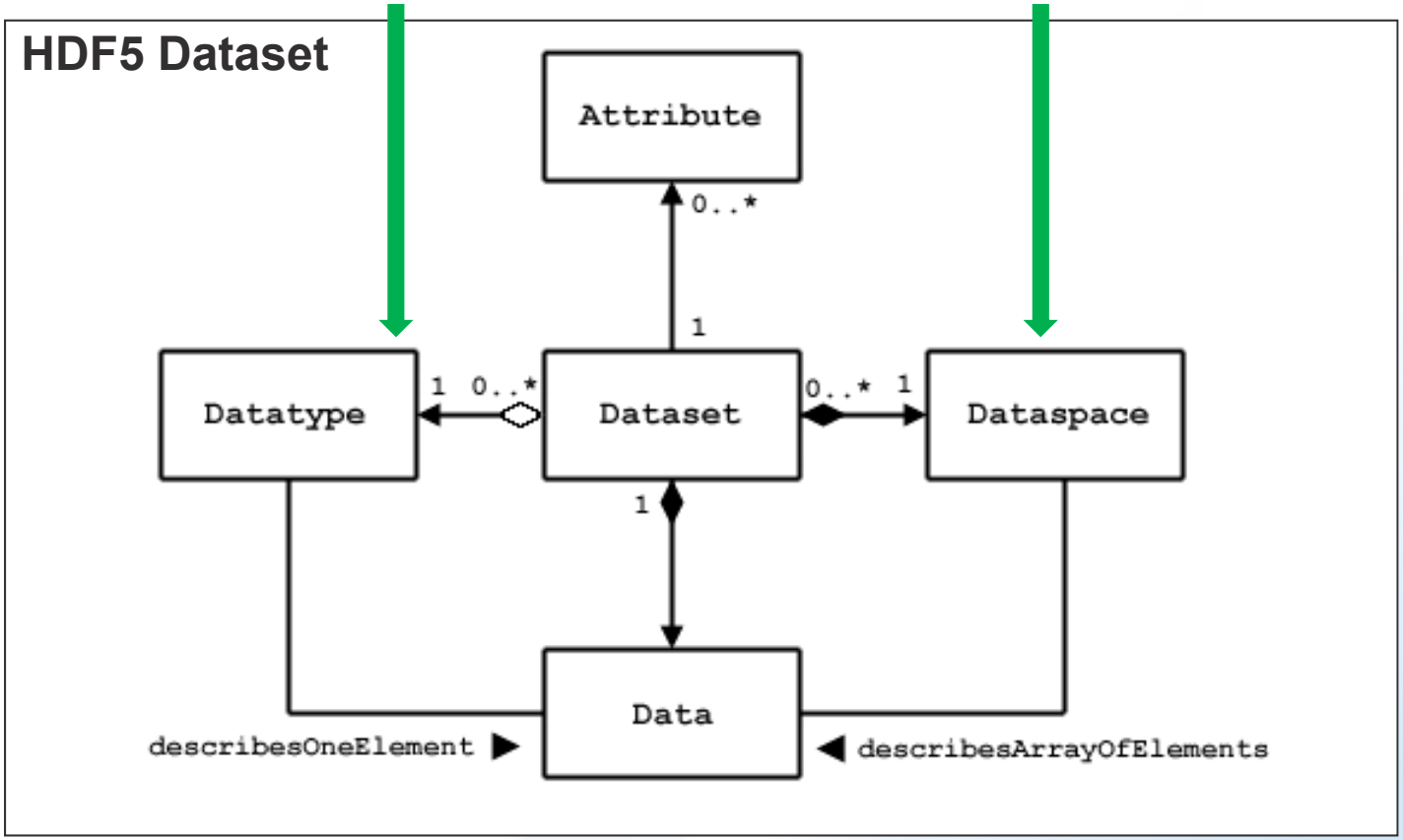
SciDB

http://www.scidb.org

- **Consistent array-oriented model and formalism :**
  - Generalization of the OLAP models
  - Natural and fundamental data type of scientific software (e.g. MATLAB)
- **Strong team of database experts lead by Mike Stonebraker**
- **Wide range of domains represented by Science Advisory Board**
  - Genomics
  - Astronomy
  - Environmental Observing Systems
  - Earth Science
  - Fusion
  - Remote Sensing
  - High Energy Physics
  - Atmospheric Sciences
  - Oceanography
  - **Control System (May, 2011)**

SciDB Array vs HDF5 Dataset

CREATE ARRAY Example < a1:integer, a2:float, a3:MyType> [Dim1=0:5, Dim2=0:4]



1. Arrays of tuples are partitioned into arrays of elements (columns)
2. Arrays of elements are partitioned into chunks

SciDB Array Processing

Array Language:

- **Array Query Language (AQL):** a set of the SQL-like commands for defining and manipulating arrays
- **Array Functional Language (AFL):** a functional language that provide the same capabilities as AQL but with a functional syntax. AFL also provide additional array operators to compose queries or statements.

Query Building Blocks:

- **Operators (e.g. join):** take one or more arrays as input and return an output array
- **Functions (e.g. sqrt):** take scalar values from literals or arrays and return a scalar value
- **Data types:** classes of values that SciDB can store and perform operations on
- **Aggregates:** take an arbitrarily large set of values as input and return a scalar value

Any of these building blocks can be **user-defined.**

Similar approach:

- **SciQL*:** SQL query language with arrays as first class citizens

---

M.Kersten, *et al.* AD 2011

SS-DB: A Standard Science DBMS Benchmark*

Four phases of scientific data management:

1. **Raw data ingest**
   - 400 2-D arrays taken at distinct time and arranged into 20 cycles. The world coordinate system goes from 0 to 10^8. Each raw image starts at (I,J) and has increasing integer values for both dimensions, ending at (I+7499, J+7499). Each array cell has 11 32-bit integers. The total benchmark size is 0.99 TBytes.
2. **Queries to the raw data**
   - User-defined functions sequencing through the 2-D arrays values and producing a collection of observations consisting of a center, polygon boundary, two observation-specific values.
3. **Cooking the raw data into a derived data set**
   - User-defined function performing a simple clustering algorithm for placing observations into groups
4. **Queries to the cooked data**
   - Q1-Q3 queries and recooking on the raw data (aggregation of values for a given slab, recooking with the different clustering function, regridding with some interpolation function),
   - Q4-Q6 queries on the observation data (aggregation of observation values for a given slab, finding polygons of observations for a given slab, tiling)
   - Q7-Q9 queries on the observation groups (finding a centroid and a sequence of centers)

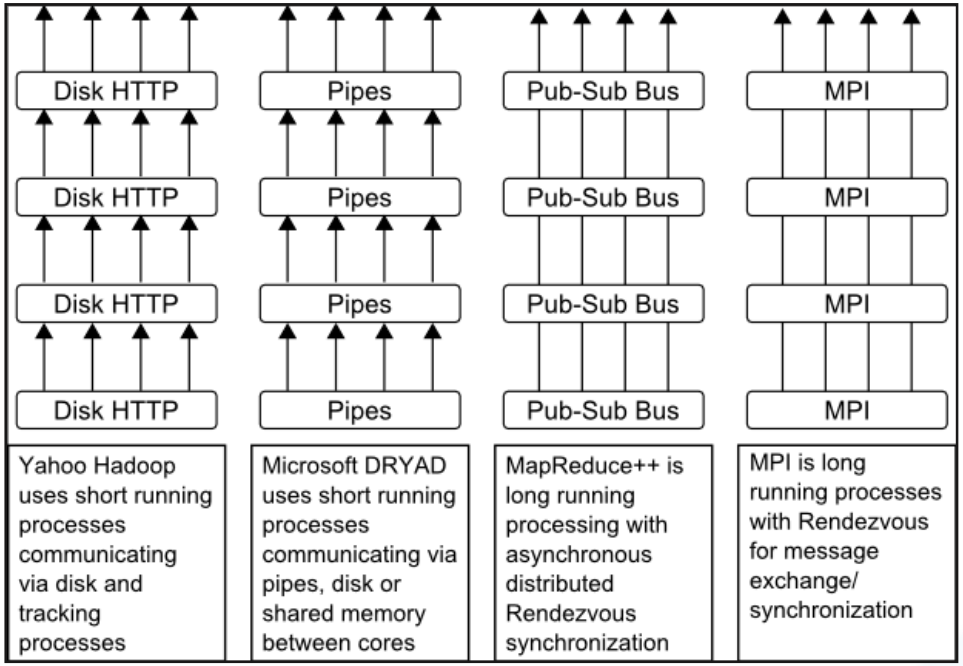
---

*P. Cudre-Mauroux *et al*. (submitted for publication)

MapReduce, Extensions, MPI

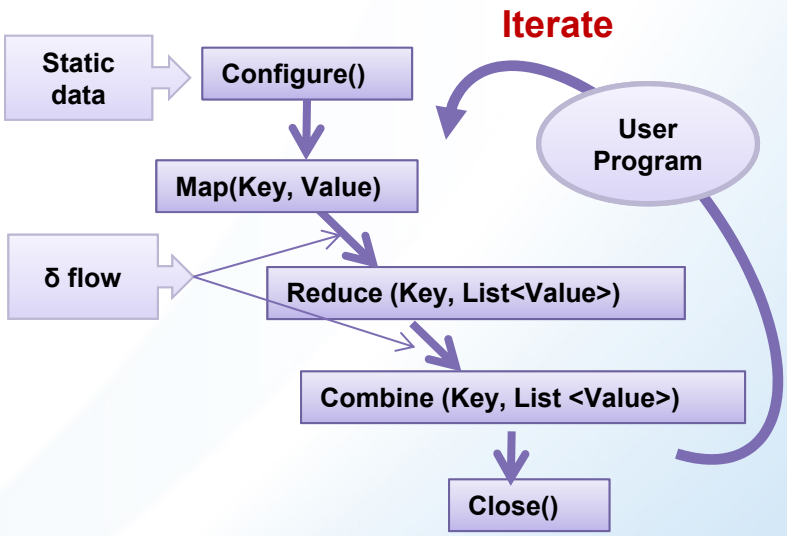
Geoffrey Fox, MPI and MapReduce, CCGSC 2010

Different synchronization and intercommunication mechanisms used by the parallel runtimes



Iterative Extensions:

- Twister [1]
- HaLoop[2]



[1] **J.Ekanayake *et al.***, HPDC 2010

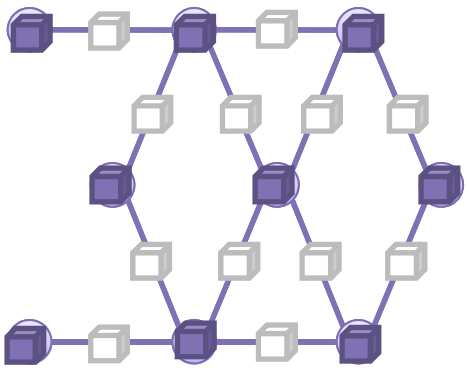
[2] **Y.Bu *et al.***, VLDB 2010

GraphLab Framework for Machine Learning

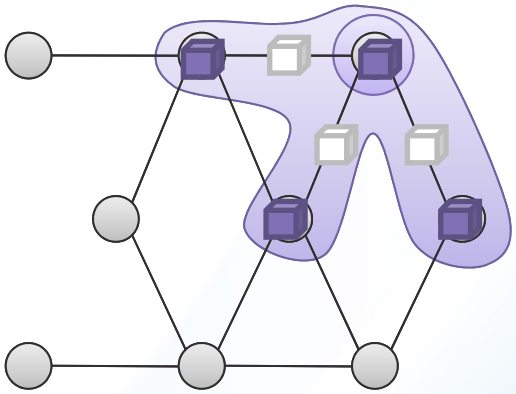
Carlos Guestrin, 1st GraphLab Workshop, July, 2012

http://graphlab.org

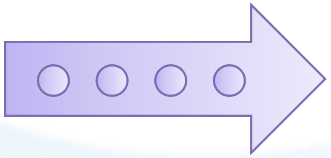
Graph Based Data Representation



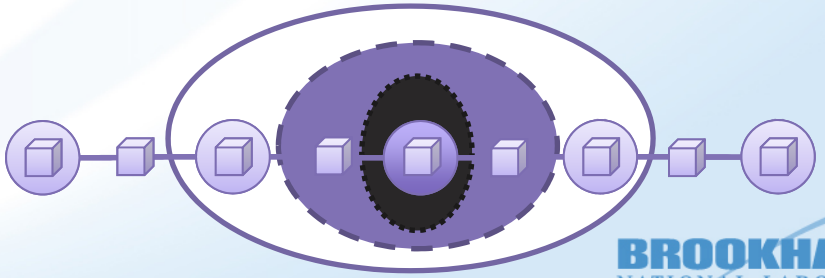
Update Functions User Computation



Scheduler



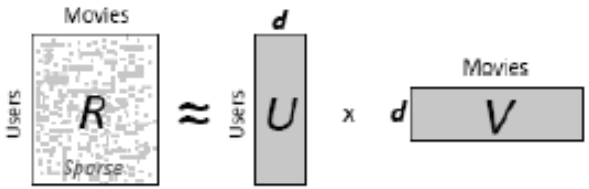
Consistency Model



Case Study: collaborative filtering

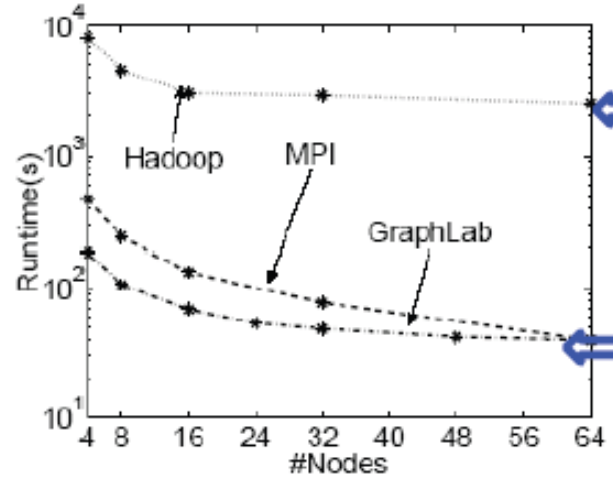
Danny Bickson, GeekSessions, 2011

- Computing a linear model for data



- Implemented alternating least square using GraphLab
- Amazon EC2 runtime results using Netflix data (sparse matrix with 100M non-zeros)

better



mahout

GraphLab Carnegie Mellon

GraphLab-based Applications …

Carlos Guestrin, 1st GraphLab Workshop, July, 2012

Alternating Least Squares

Splash Sampler

SVD

Bayesian Tensor Factorization

Lasso

CoEM

LDA

PageRank

Belief Propagation

SVM

Gibbs Sampling

Dynamic Block Gibbs Sampling

K-Means

…Many others…

Matrix Factorization

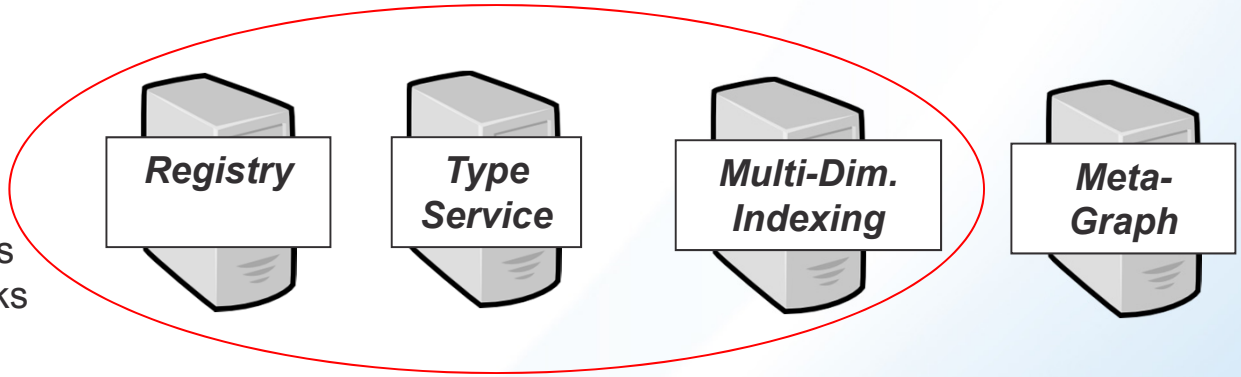
Linear Solvers

In-situ Approach for Processing the HDF5 Data with the Large-Scale Analytics Engines

Different types of the distributed analytics systems
(e.g. shared-nothing, shared-disk, etc.)

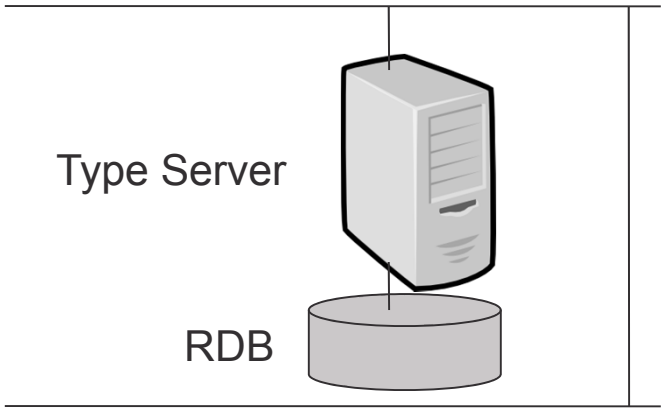
Distributed Services of the HDF5 Model:

- User-Defined Data Types
- Multi-Dimensional Datasets
- Groups of Hierarchical Links



Distributed HDF5 files

Type Server



RDB representation of
the OMG DDS Extensible and Dynamic Topic Types Specification

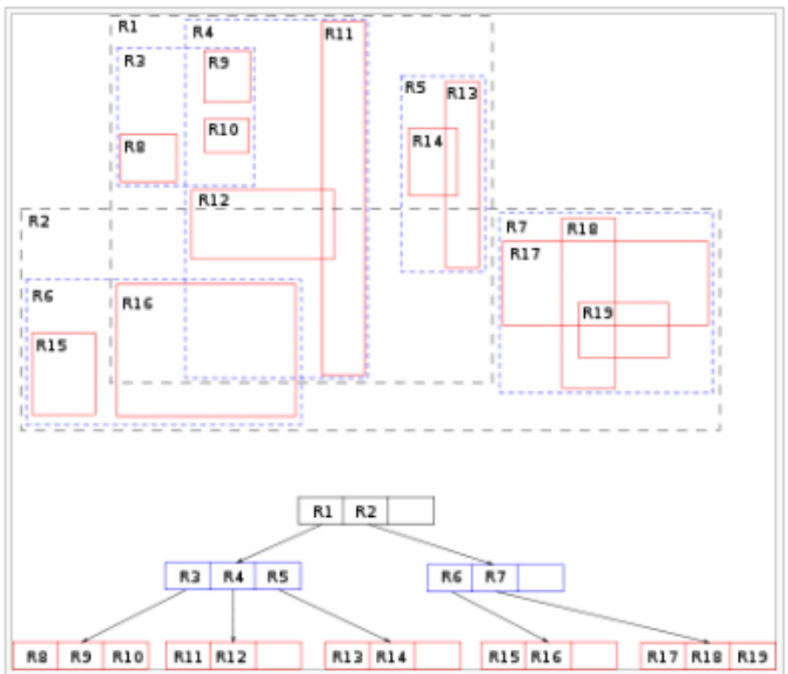
- **Type System:** model of the data types defined in UML, independent
- **Type Representation:** ways in which types may be externalized, such as IDL, XSD, XML, Type Object (binary) **+ RDB**
- **Data Representation:** ways in which objects of the types may be externalized
- **Language Binding**: ways in which applications can access the state of objects

Task Force Chair: Rick Warren
Final Release: 2012

Revision / Finalization Task Force Membership

Member	Organization
Angelo Corsaro	PrismTech
Dario Di Crescenzo	Selex
Charlie Fudge	Naval Surface Warfare Center (NSWC)
Sam Mancarella	Sparx Systems
Nikolay Malitsky	Brookhaven National Laboratory
Ken Rode	Gallium Visual Systems
John Thier	General Dynamics AIS
Char Wales	MITRE
Rick Warren	Real-Time Innovations (RTI)
Virginie Watine	Thales
Johnny Willemson	Remedy IT
Chuck Zublic	Northrop Grumman

Indexing of the Multidimensional Datasets



Simple example of an R-tree for 2D rectangles

Wikipedia, http://en.wikipedia.org/wiki/R-tree

Antonin Guttman (1984), R-Trees: A Dynamic Index Structure for Spatial Searching.

B.Nam and A.Sussman (2004), A Comparative Study of Spatial Indexing Techniques for Multidimensional Scientific Datasets.

Historical Datasets

- **C. Kolovson and M. Stonebraker (1989)**, Indexing Techniques for Historical Databases.
- **S. Chevtsov (2004),** EPICS Channel Archiver.

HDF5-related projects:

- **B. Nam and A. Sussman (2003),** Improving Access to Multi-dimensional Self-Described Scientific Datasets.
- **L. Gosink et al. (2006),** HDF5-FastQuery: Accelerating Complex Queries on HDF Datasets using Fast Bitmap Indices.

Large-Scale Systems:

- **H.Liao, J.Han, J.Fang (2010)**, Multi-dimensional Index on Hadoop Distributed File System.
- **B. Nam and A. Sussman (2011)**, Analyzing Design Choices for Distributed Multidimensional Indexing.

Integrated System