paradigm4

data-driven discovery

Scientific DBMSs at Scale and SciDB

Mike Stonebraker

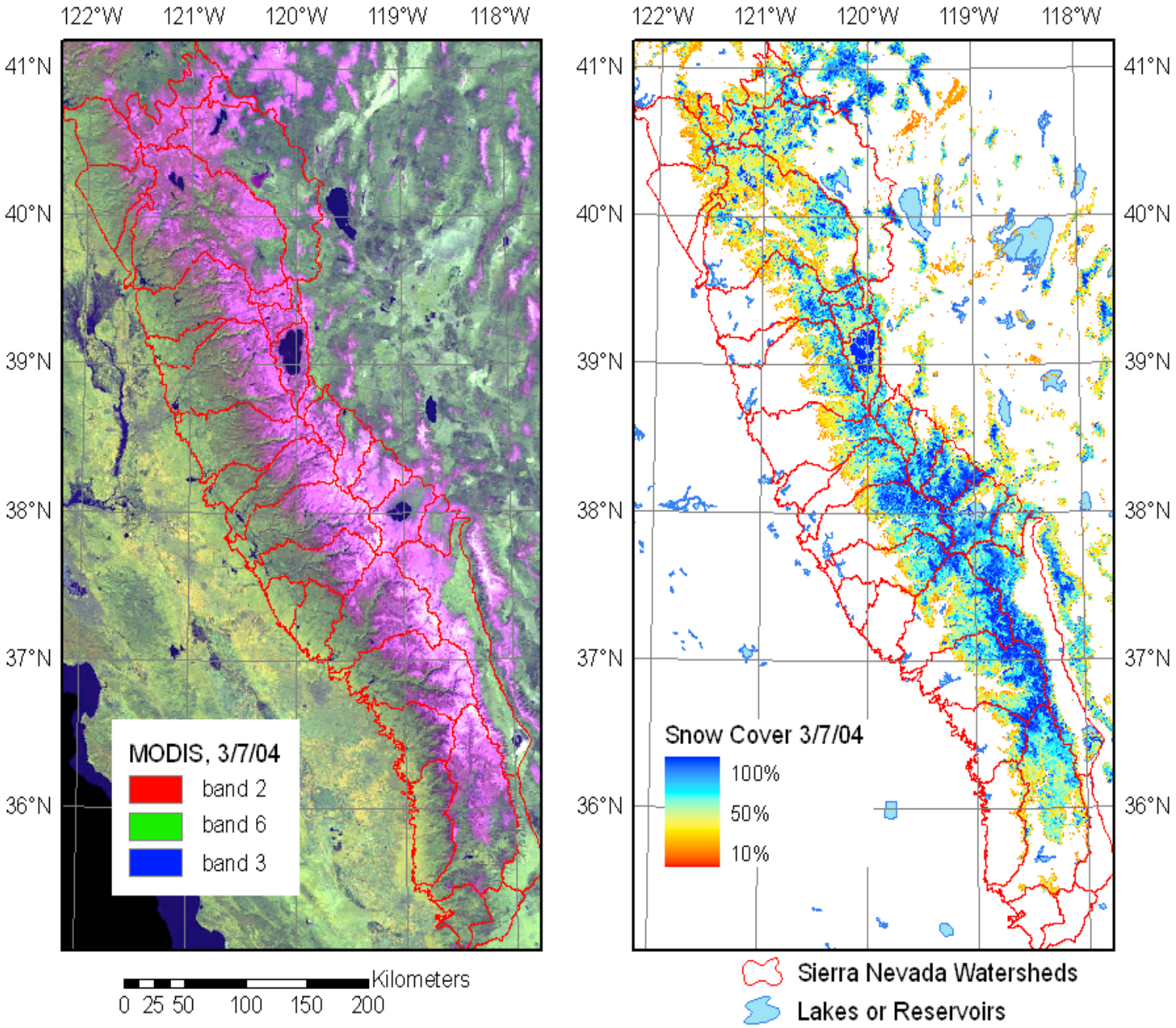
Outline

- Data architecture
  - File system
  - RDBMS
  - other
- Support for analytics

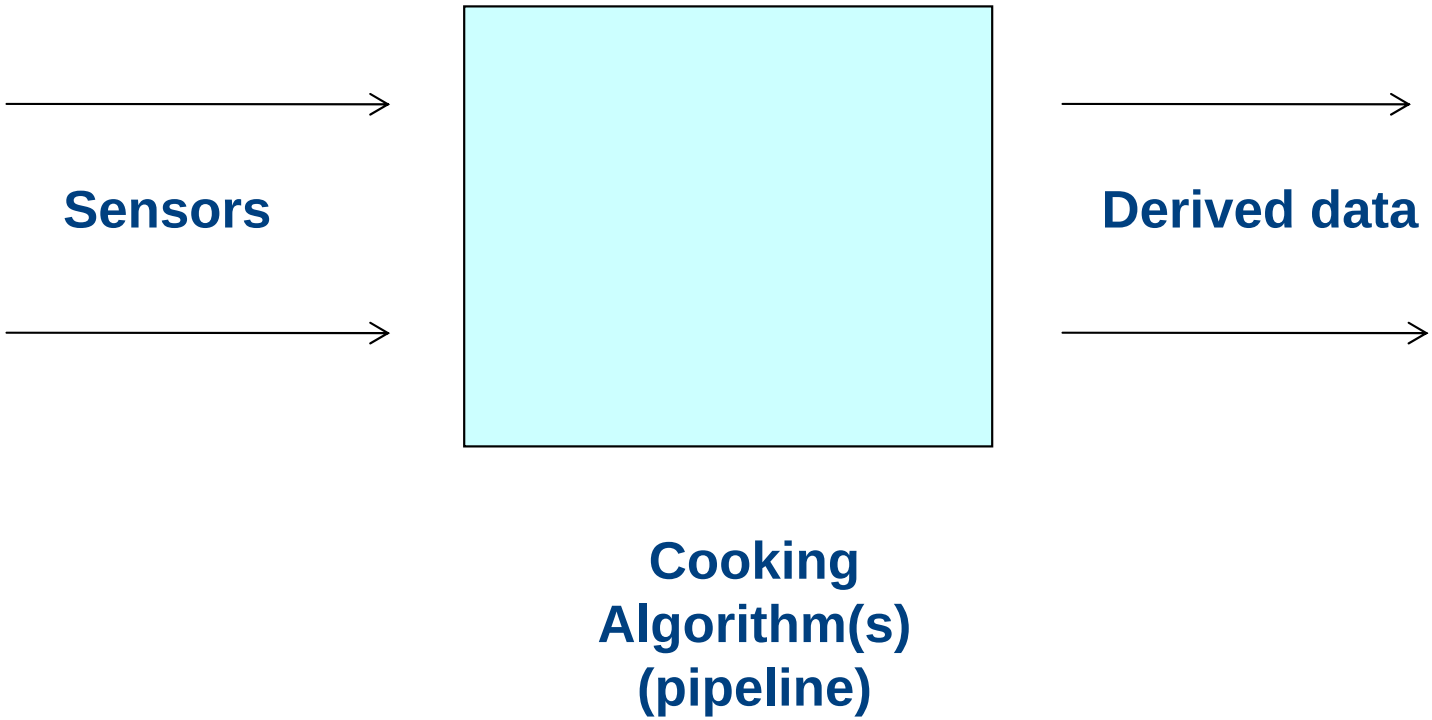
Example Application -- MODIS

- I know nothing about HEP, so you get satellite imagery
- 2 satellites
  - Trace a “wide piece of scotch tape” continuously around the earth
  - At multiple frequencies
- Cooked into several levels of data products by NASA
  - For example, “best cloud free cell” from multiple passes
- Baked into science metrics by users
  - For example snow cover

Snow Cover in the Sierras



General Model



Traditional Wisdom (1)

- Cooking pipeline in hard code or custom hardware
- All data in some file system

Problems

- Can’t find anything
  - No query language
  - No schema
- Metadata often not recorded
  - Sensor parameters
  - Cooking parameters
- Can’t easily share anything
  - Have to export your access programs
- Can’t easily recook anything
  - Big problem with MODIS
- Everything is custom code
  - Supported by an army of Postdocs

Traditional Wisdom (2)

- Cooking pipeline outside DBMS
- Derived data loaded into RDBMS for subsequent querying
- Used by Sloan Sky Survey

RDBMS Issues

- Pretty hopeless on raw data
  - Simulating arrays on top of tables likely to cost a factor of 10-100
- Not pretty on time series data
  - Find me a sensor reading whose average value over the last 3 days is within 1% of the average value adjoining 5 sensors
- Not pretty on spatial data
  - Find me snow cover within 10 miles of Mt. Whitney

RDBMS Summary

- Wrong data model
  - Arrays not tables
- Wrong operations
  - Regrid not join
- Missing features
  - Versions, no-overwrite, provenance, support for uncertain data, …

But your mileage may vary

- SQL Server working well for Sloan sky survey database
  - Only derived data….
- See paper in CIDR 2009 by Jose Blakeley

My Preference

- Load the raw data into a DBMS
- Cooking pipeline is a collection of user-defined functions (DBMS extensions)
- Activated by triggers or a workflow management system
- ALL data captured in a common system!!!!

In Other Words…..

- A “data centric” architecture
- To replace the “compute centric” HPC architectures that are the norm today
- And run on commodity hardware

System Requirements

- Complex analytics
  - E.g snow cover
- Data management
  - Cooking
  - Leave out outliers
  - Queries on derived data
- Scalability to:
  - many cores
  - many nodes
  - out-of-memory data

What Analytics Do You Need?

an accessible example

- Consider the closing price on all trading days for the last 20 years for two stocks A and B
- What is the covariance between the two time-series?

$$(1/N) * \text{sum} (A_i - \text{mean}(A)) * (B_i - \text{mean} (B))$$

Now Make It Interesting …

Do this for all pairs of 15000 stocks

- The data is the following 15000 x 4000 matrix

Stock	t_1	t_2	t_3	t_4	t_5	t_6	t_7	….	t_{4000}
S_1									
S_2									
…									
S_{15000}									

Array Answer

Ignoring the (1/N) and subtracting off the means ….

$$Stock * Stock^{T}$$

Yabut…..

- Array has 60M cells (easily fits in main memory)
- Multiply complexity is (15000) * (15000) * (4000) floating point operations……
  - .9 Teraflop

- What about hourly data (X8) or tick-level data?
- What about all stocks (X4)?
- What about high, low, volume, …?
- What about bid-ask data?
- What about options?

Gets big in a hurry!

These Requirements Arise in Many Non-Science Domains

- Recommendation engines
  - People who liked XXX also liked YYY
  - Clustering customers in a high dimensional space is one popular technique
- Predicting unscheduled down-time in complex machinery
  - Oil refineries, jet engines, helicopters, ….
  - Predictive modeling in high dimensional spaces

Solution Options

- SAS, R, S, SPSS, HDF5…
  - Weak or non-existent data management
- RDBMS
  - Weak or non-existent linear algebra
- 2 Systems
  - Learn 2 systems, and copy the world back and forth
- Hadoop
  - Good only at “embarrassingly parallel” tasks
  - Hit the wall the minute you try to scale

Better Answer: An Array DBMS

e.g. SciDB from Paradigm4

- All-in-one: data management with massively scalable advanced analytics
- Data is updated via time-travel; not overwritten
  - Supports reproducibility for research and compliance
- Supports uncertain data, provenance
- Open source
  - Developed and supported by Paradigm4
- Hardware agnostic
  - Runs on commodity hardware clusters or cloud

Native N-Dimensional Array Data Model

optimal for data with intrinsic ordering

- Array cells can contain many values and types of values (*attributes*)
- High performance support for both sparse and dense data
- Support for non-integer dimensions
- User-definable, context-sensitive missing code substitution
  - Support for 256 null or missing codes (as opposed to SQL 3VL null)

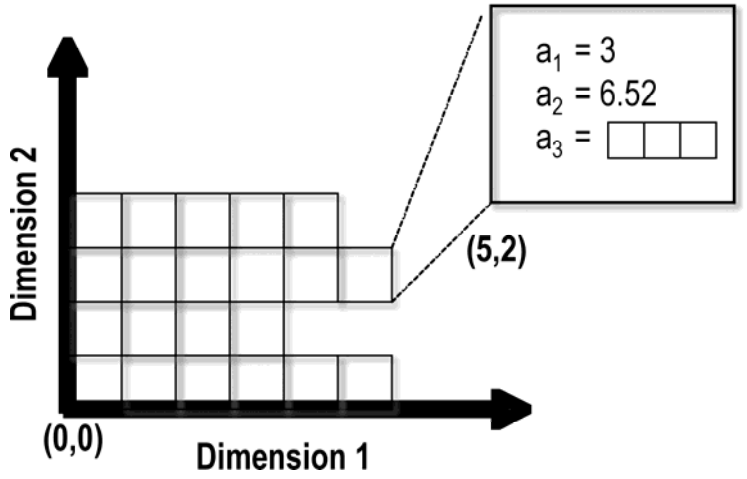
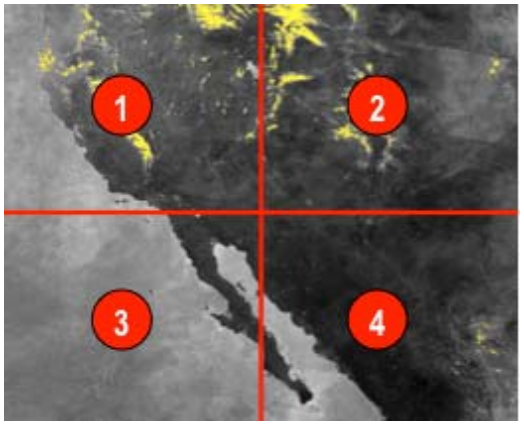


Illustration is 2D, but data can have unlimited dimensions

SciDB native array model




```
CREATE ARRAY RGB
  < red     : int16,
    green   : int16,
    blue    : int16>
 [ longitude(double) = *, 10000, 0,
   lattitude(double) = *, 10000, 0 ];
```

Attributes	Dimensions	Dimension size	Chunk	Chunk
red, green, blue	longitude, lattitude	* indicates unbounded	size	overlap

Arrays are chunked …. with optional overlap

Chunk 1

0.02	0.01	0.01	0.02
0.01	0.01	0.5	0.02
0.01	0.02	0.01	0.01
0.02	0.01	0.02	0.02

Chunk 2

0.02	0.01	0.01	0.02
0.01	0.01	0.5	0.02
0.01	0.02	0.01	0.01
0.02	0.01	0.02	0.02

Chunk 3

0.02	0.01	0.01	0.02
0.01	0.01	0.5	0.02
0.01	0.02	0.01	0.01
0.02	0.01	0.02	0.02

Chunk 4

0.02	0.01	0.01	0.02
0.01	0.01	0.5	0.02
0.01	0.02	0.01	0.01
0.02	0.01	0.02	0.02

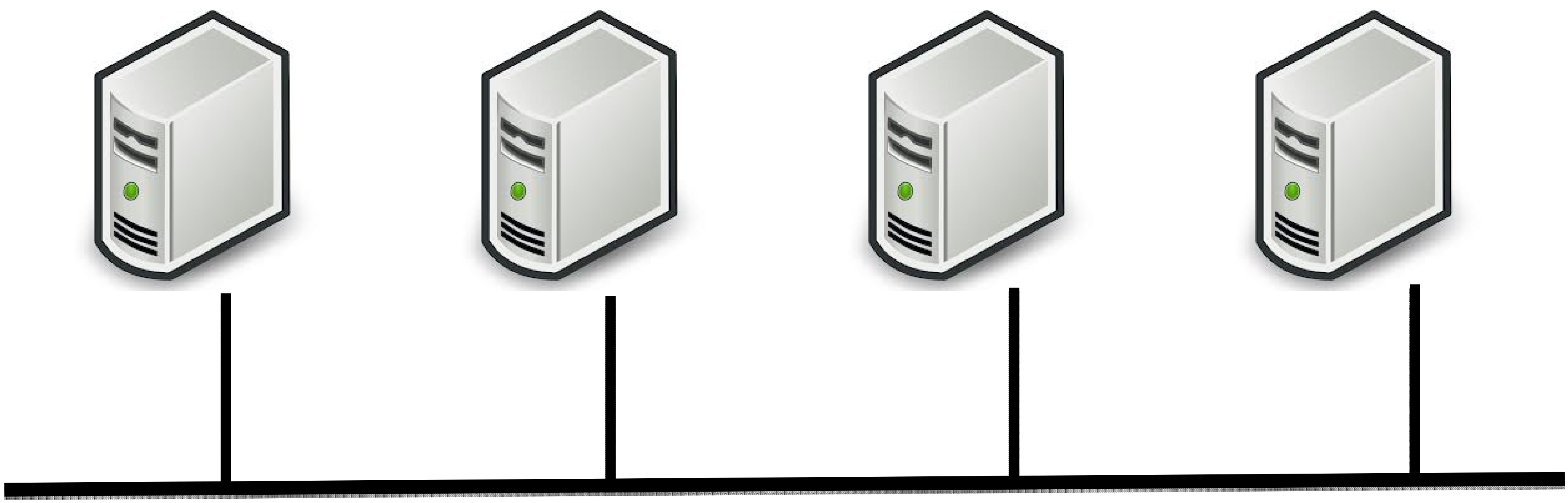
Data are distributed across SciDB instances

0.02	0.01	0.01	0.02
0.01	0.01	0.5	0.02
0.01	0.02	0.01	0.01
0.02	0.01	0.02	0.02

0.02	0.01	0.01	0.02
0.01	0.01	0.5	0.02
0.01	0.02	0.01	0.01
0.02	0.01	0.02	0.02

0.02	0.01	0.01	0.02
0.01	0.01	0.5	0.02
0.01	0.02	0.01	0.01
0.02	0.01	0.02	0.02

0.02	0.01	0.01	0.02
0.01	0.01	0.5	0.02
0.01	0.02	0.01	0.01
0.02	0.01	0.02	0.02



P4's native Array DB beats Relational DBs*
on storage efficiency & complex computations

Relational Database

I	J	value
0	0	32.5
1	0	90.9
2	0	42.1
3	0	96.7
0	1	46.3
1	1	35.4
2	1	35.7
3	1	41.3
0	2	81.7
1	2	35.9
2	2	35.3
3	2	89.9
0	3	53.6
1	3	86.3
2	3	45.9
3	3	27.6

48 cells

Array Database

32.5	46.3	81.7	53.6
90.9	35.4	35.9	86.3
42.1	35.7	35.3	45.9
96.7	41.3	89.9	27.6

16 cells

- 10-100x faster than RDBMSs on array operations on both sparse and dense data
- Math operations run natively directly in the database
- Dramatic storage efficiencies as # of dimensions & attributes grows

*** Applies to both row stores & column stores**

Array Query Language (AQL)

- Array data management
  - e.g. filter, aggregate, join, etc.
- Statistical & linear algebra operations
  - multiply, QR factorization, etc.
  - parallel, disk-oriented
- User-defined operators (Postgres-style)
  - Snow cover

Array Query Language (AQL)

```
SELECT Geo-Mean ( T.B )
FROM Test_Array T
WHERE
   T.I BETWEEN :C1 AND :C2
AND T.J BETWEEN :C3 AND :C4
AND T.A = 10
GROUP BY T.I;
```

User-defined aggregate on an attribute B in array T

Subsample

Filter

Group-by

Client interfaces

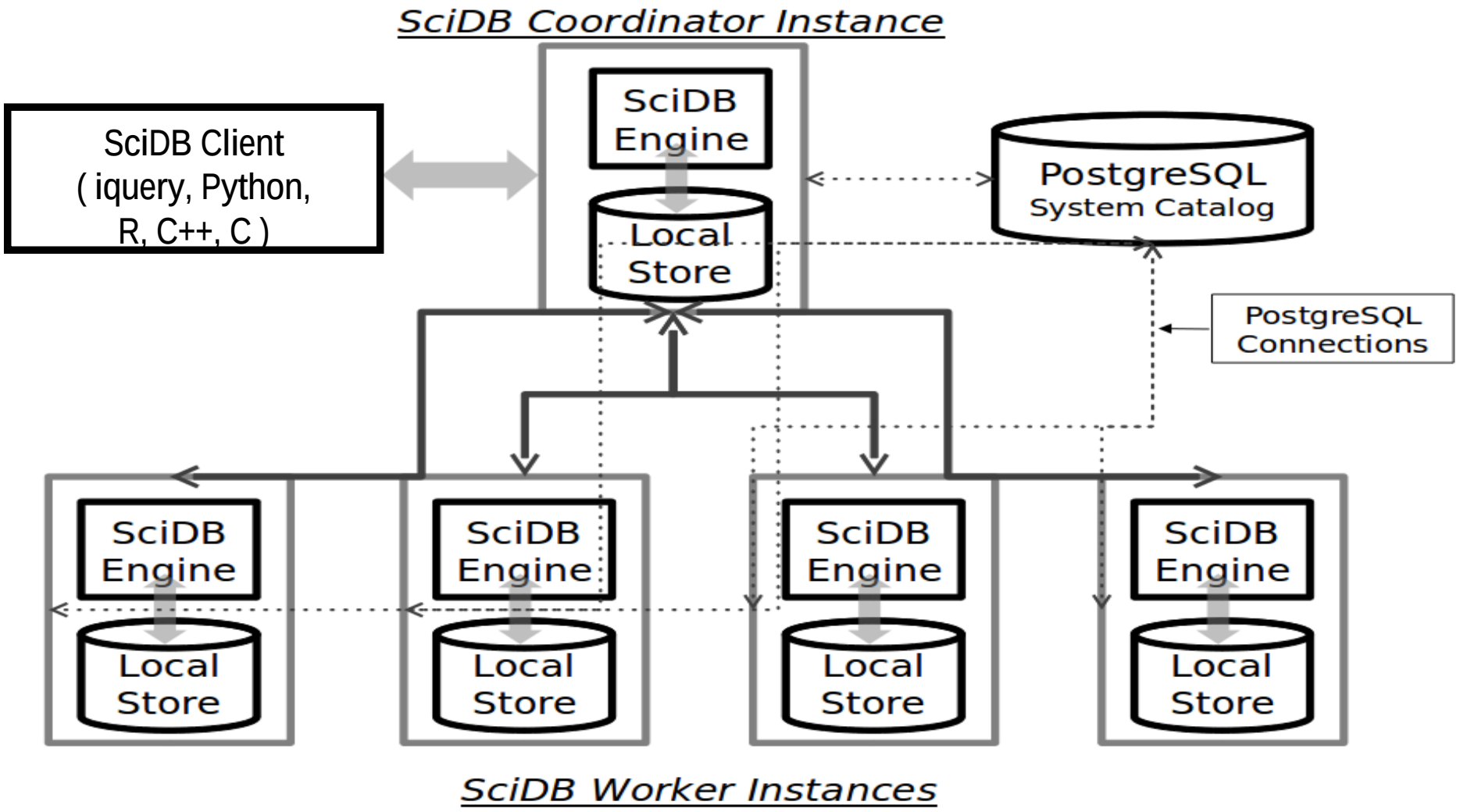
- i-query interactive command line query interface
- C++
- Python
- R
- Java / JDBC

SciDB-R
an R front end for SciDB

- Develop within a familiar R IDE (including R Studio)
- Direct access from a new R array class to data stored in SciDB
  - Arrays not limited by standard R array indexing limits
  - Data materialized only when requested, minimizing data movement
- Transparently offload large computations to SciDB
  - Linear algebra operators and matrix decompositions are overloaded and run in parallel in SciDB
  - Different approach than other databases with multiple R instances running independently on each node from UDFs

- Higher performance with much less code than using R with a separate database or data from files

SciDB System Architecture

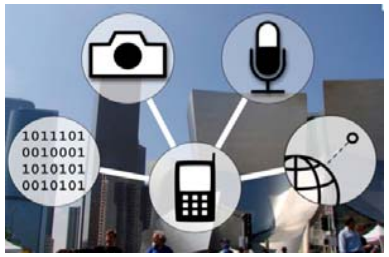


- "Shared Nothing" cluster of commodity hardware nodes
- Interconnected with standard ethernet and TCP/IP

Status and Performance

- SciDB is 100x Postgres on array-like queries
- SciDB is faster or the same on vanilla data management
- SciDB is comparable to R on analytics
  - But scales!
  - And provides shared access to data with ACID guarantees

Broad range of early adopters



Commercial Companies

- e-commerce
- Pharma
- Insurance
- Computational Finance

Scientific

- NCBI One Thousand Genomes project
- Lawrence Berkeley National Labs
- NASA Goddard
- EarthDB (Frew)

Prediction

- Science will move from roll-your-own
  - To DBMSs
  - Just like commercial users did long ago
- HPC will have to include data management
  - Not just CPU
  - Will force them away from “compute farms”
- Science
  - Is a zero billion $$ market
  - But will piggyback on commercial users
  - With huge upside

More Generally….

- As the world moves from simple analytics to complex analytics
  - RDBMSs likely to fail
  - And Hadoop unlikely to scale
- Check out SciDB
  - Download from **www.scidb.org/forum**
  - Register for next SciDB-R webinar on 6/6/13