



Big Data Management with the Myria Cloud Service

Magdalena Balazinska

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

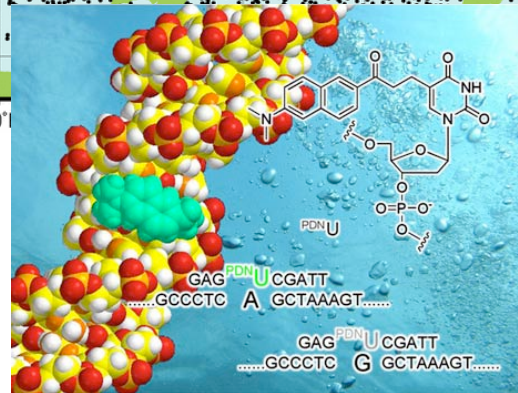
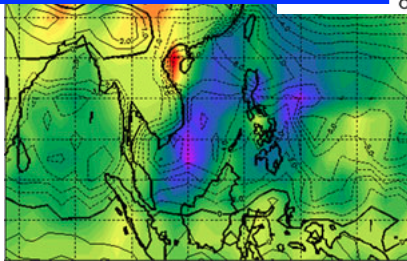
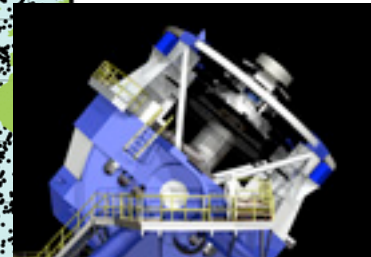
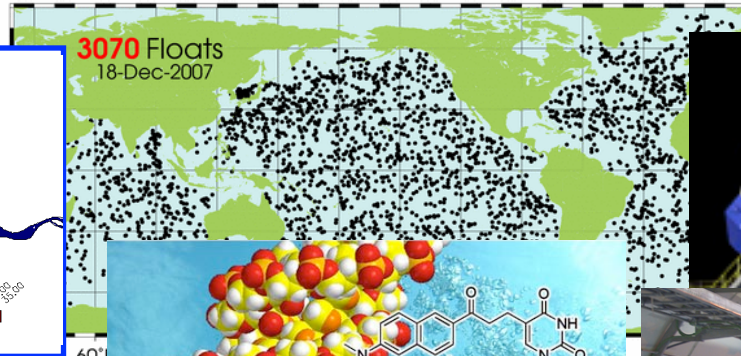
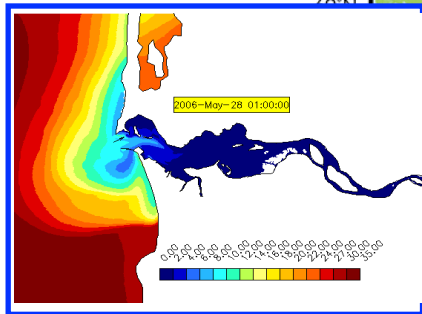
UNIVERSITY OF WASHINGTON

<http://www.cs.washington.edu/people/faculty/magda>



Science is Facing a Data Deluge!

- **Astronomy:** High-resolution, high-frequency sky surveys (SDSS, LSST)
- **Medicine:** ubiquitous digital records, MRI, ultrasound
- **Biology:** lab automation, high-throughput sequencing
- **Oceanography:** high-resolution models, cheap sensors, satellites
- Etc.



Example: Astronomy

How did the universe at 300,000 years

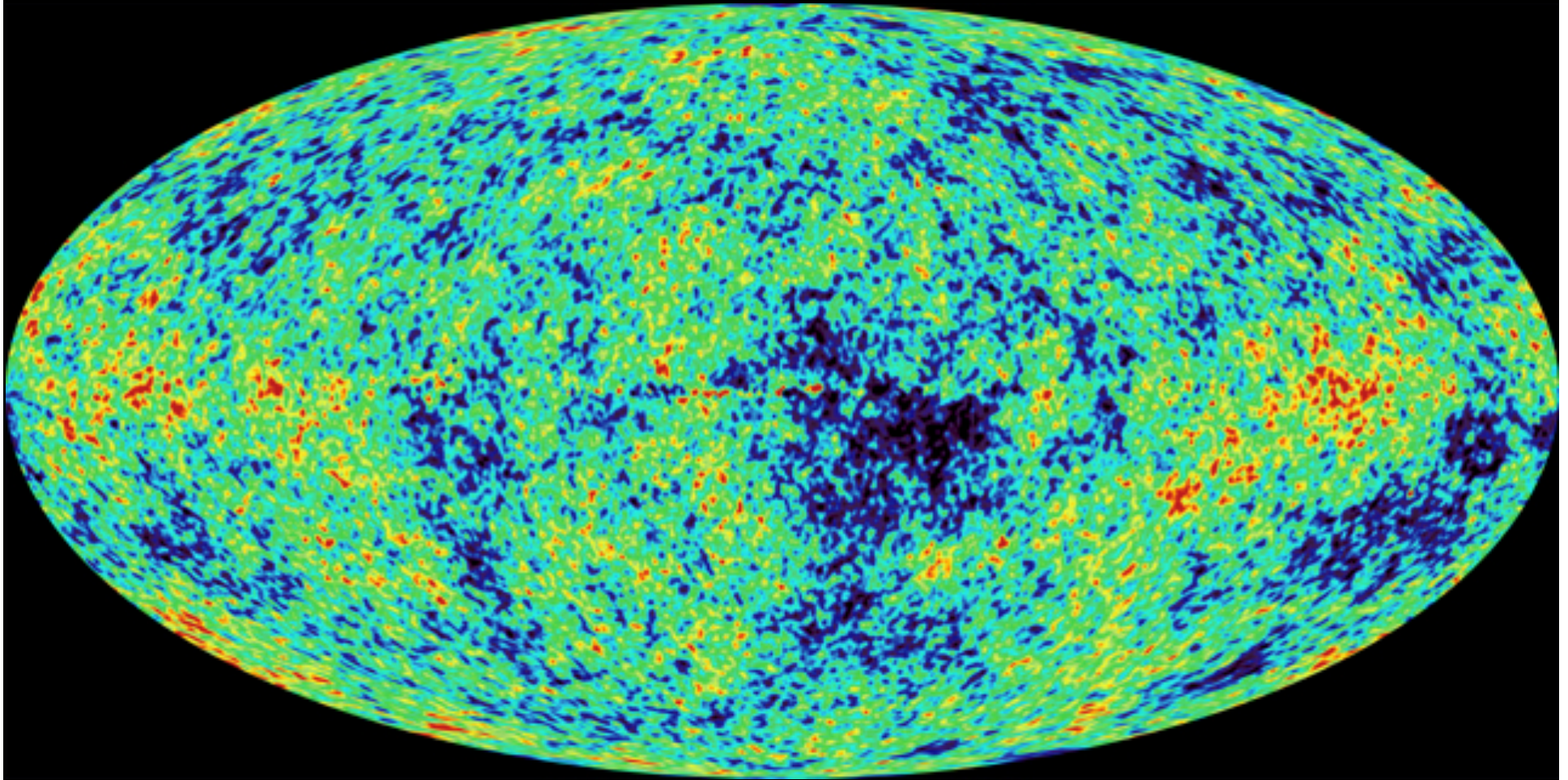


Image courtesy NASA/WMAP

Slide from Andrew Connolly (Astronomy Dept.)

... turn into this?



Slide from Andrew Connolly (Astronomy Dept.)

...and this....



Slide from Andrew Connolly (Astronomy Dept.)

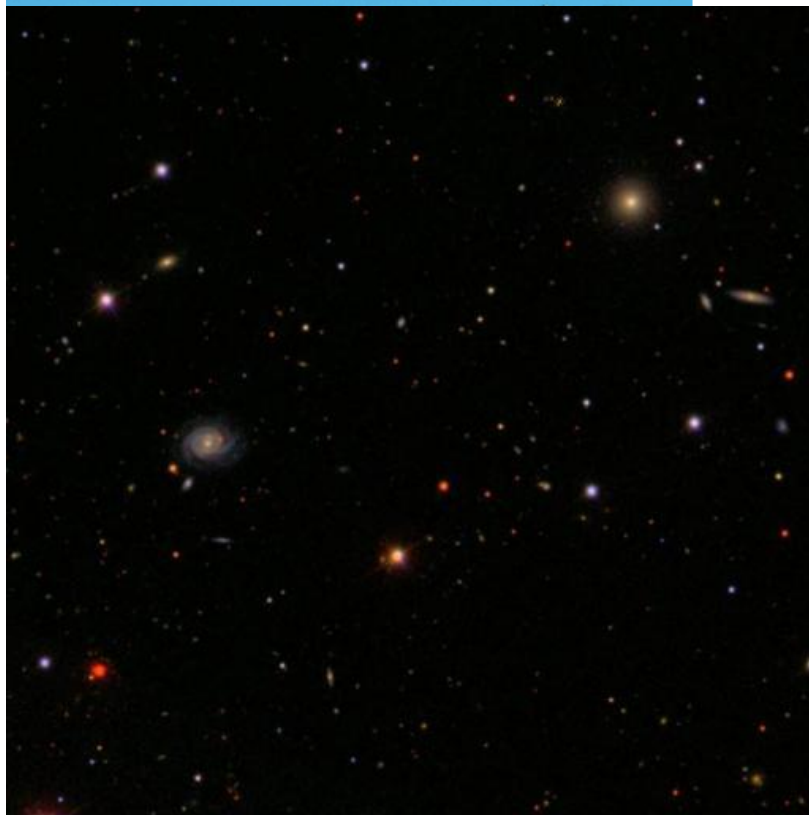
...and this?



Slide from Andrew Connolly (Astronomy Dept.)

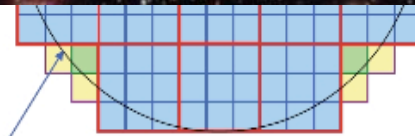
Method 1: Observe the Sky

Large Synoptic Survey Telescope

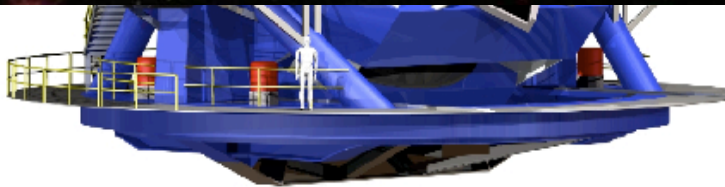


Wavefront
location

Sensor
(locations)

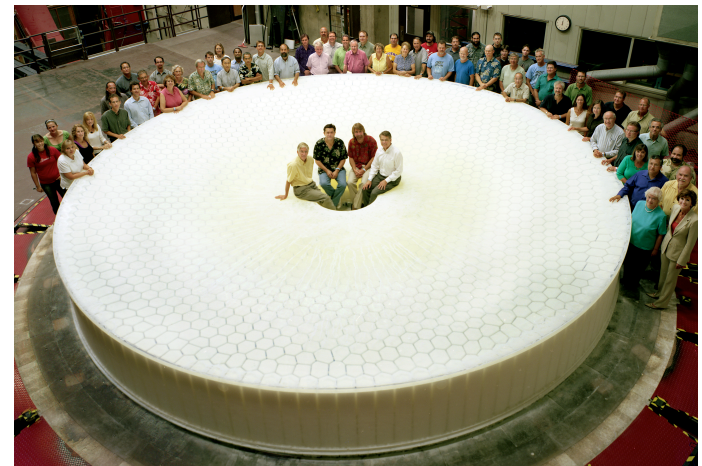


3.5 degree Field
of View (634 mm diameter)



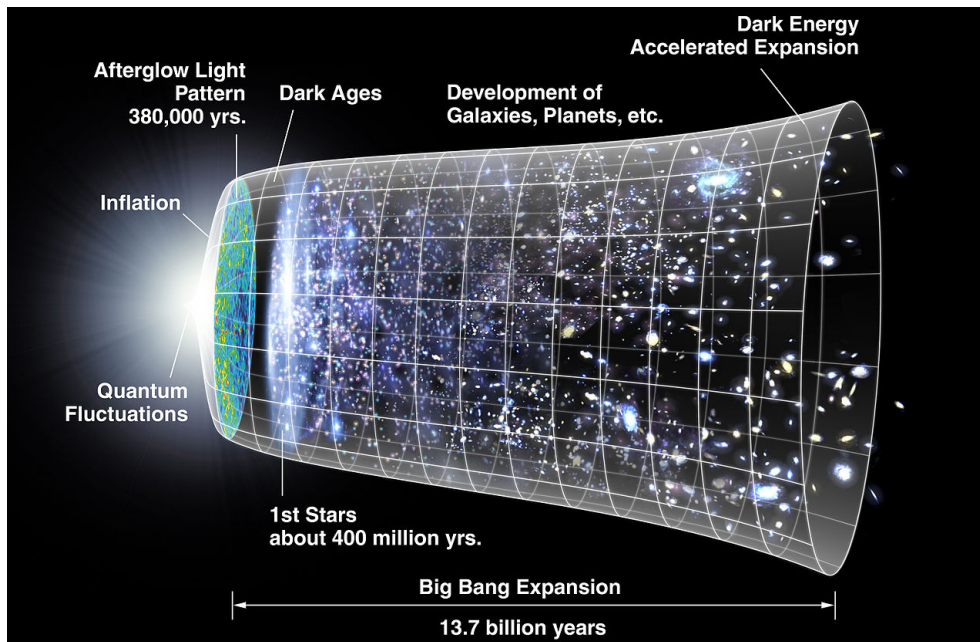
Challenges from new astronomy

- Sloan Digital Sky Survey (SDSS)
 - 7 years of imaging
 - 8000 sq degrees of the sky (1/5th)
 - 200 million stars and galaxies
 - **80 TB raw images**
- LSST data flow
 - 20,000 sq degrees every 3 nights
 - **40 TB of imaging per night**
 - 10^8 sources a night (10^3 “events”)
 - 1000 repeat observations over 10 years
 - 10 Petabytes of catalogs (10 years)
 - 100 PBs of images
 - 5 months to watch 1 year of data (HDTV)
 - Data public as soon as taken



Method 2: Simulate

- Evolution of large-scale structure in the universe
 - Universe is a set of particles (gas, dark matter, stars)
 - Output snapshot every few simulation timesteps



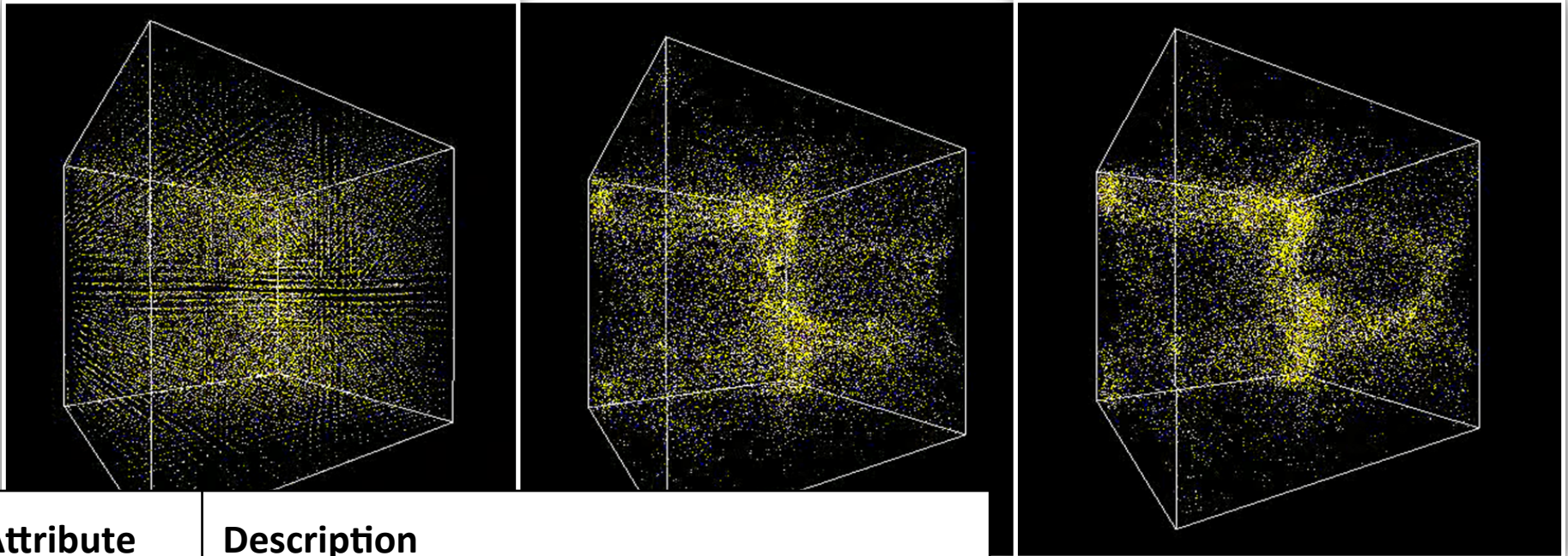
Typical simulations today

Spring 2014: 5TB

Fall 2014: 200TB

UW N-body
group

N-Body Simulations Data

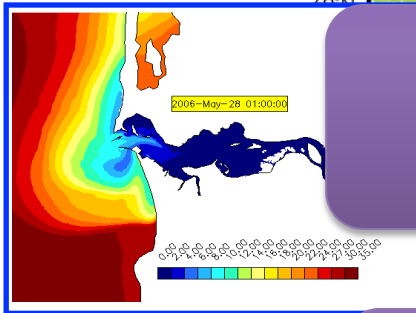


Attribute	Description
iOrder	unique identifier
X, Y, Z	position in Cartesian coordinates
Type	type of particle: either dark, star or gas
Grp	halo group particle belongs in
Time	Timestep the particle belongs in

Universe simulation
46 snapshots
5 TB

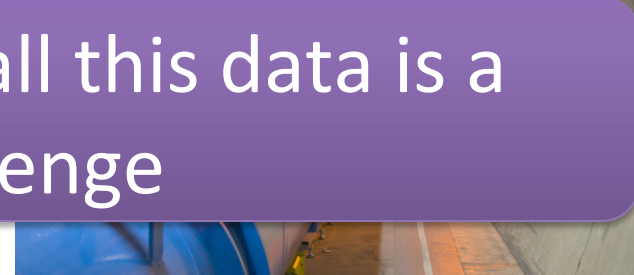
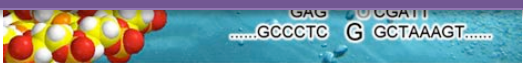
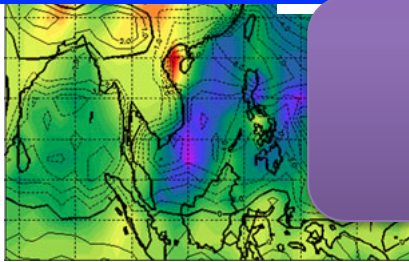
Big Data Need in All Sciences!

- **Astronomy**: High-resolution, high-frequency sky surveys (SDSS, LSST)
- **Medicine**: ubiquitous digital records, MRI, ultrasound
- **Biology**: lab automation, high-throughput sequencing
- **Oceanography**: high-resolution models, cheap sensors, satellites
- Etc.



Data holds the promise to accelerate discovery

But analyzing all this data is a challenge

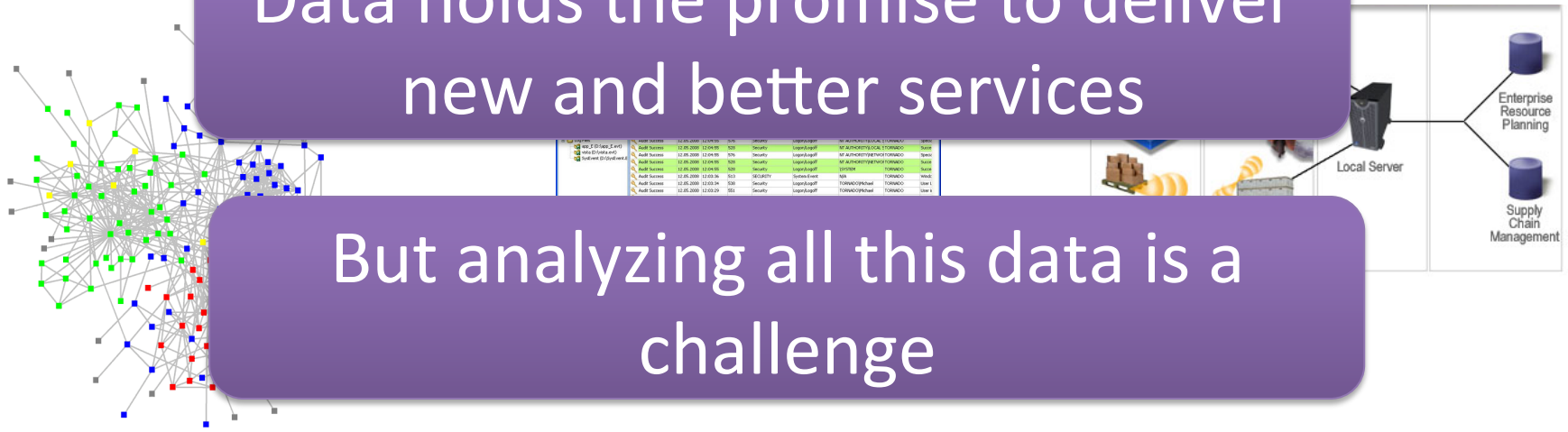


Industry is Facing a Data Deluge!

- Clickstreams, search logs, network logs, social networking data, RFID data, etc.
- Examples: Facebook, Twitter, Google, Microsoft, Amazon, Walmart, etc.

Data holds the promise to deliver new and better services

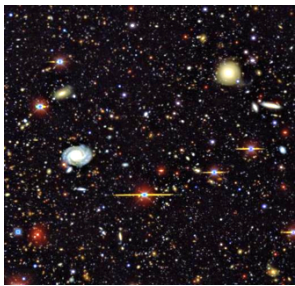
But analyzing all this data is a challenge



Challenging Application Requirements

Exciting and challenging requirements

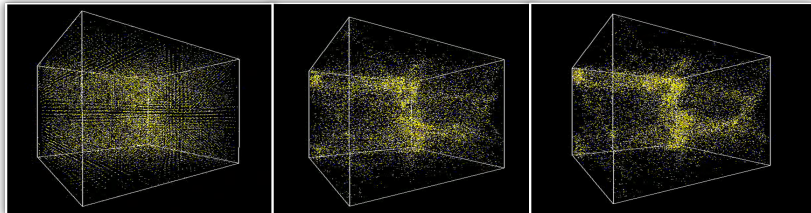
Often generalize beyond campus



Telescope image:

1. Iterative data cleaning
2. Objects extraction
3. Classification

Picture from Deep Lens Survey (DLS: Tyson)

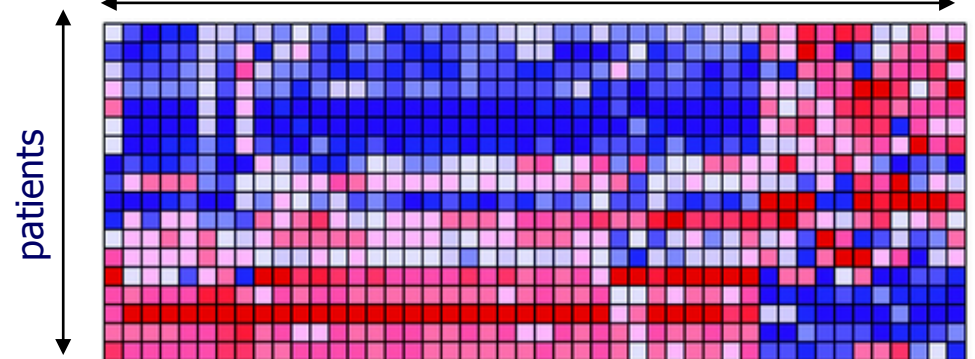


Picture from D. H. Stalder et. al. [arXiv:1208.3444](https://arxiv.org/abs/1208.3444) [astro-ph.CO]

N-body simulation data:

1. Manage hundreds of TB of data
2. Data clustering to extract galaxies
3. Graph analytics to study galaxy evolution

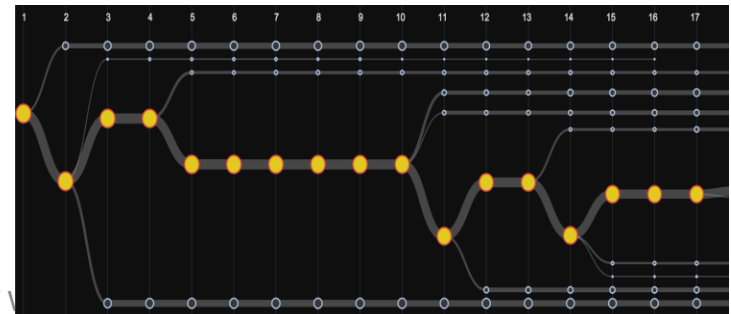
>1M features from molecular snapshot



Genome data processing:

Picture from Su-In Lee

1. Linear algebra on large matrices
2. Novel machine learning algorithms



How Can We Analyze All this Data

- Excel? Limit of 65,536 rows by 256 columns
- Write my own Java/Python/other program?
 - Not all data fits in memory
 - May want to run in parallel in a cluster
 - Do I really want to write a program for each question?
- Use an existing database management system
 - Today's data **V**olume, **V**elocity, and **V**ariety are breaking these systems... new systems are appearing
- Build a new database management system!

Myria Big Data Management Service

Myria is a Cloud service: Just open browser and go!

The screenshot displays the Myria web interface. At the top, there is a navigation bar with the Myria logo, 'Editor', 'Queries', and 'Datasets' tabs, and a status bar with 'Report an issue' and 'rest.myria.cs.washington.edu:1776 [72/72]'. The main content area is split into two panels. The left panel is a code editor with a light blue header that says 'Write your code here, perhaps starting from one of the examples at the right.' It contains a MyriaL query script with 11 lines of code. Below the code are three buttons: 'Execute the Query', 'Parse', and 'Myria JSON'. The right panel has tabs for 'Examples', 'Datasets', 'Query Plan', and 'Results'. Below the 'Query Plan' tab, it says 'Visualization of the logical and optimized physical query plan.' and 'Code parsed as Relational Algebra'. A large diagram titled 'Relational algebra converted and optimized into a Myria Physical Plan' shows a flow of operations: 'Scan(armbrustlab:seaflo...)', 'Select((\$14 = "beads"))', 'Apply(Cruise=\$11,...)', and 'GroupBy(\$0; SUM(\$1), COUNTALL, SUM(\$2), SUM(\$7), SUM(\$...))'. A red banner at the bottom of the interface contains the URL 'http://myria.cs.washington.edu'. In the bottom right corner, there is a page number '16'.

```
1 good_opp_vct = scan(armbrustlab:seaflo:good_opp_vct_v4);
2
3 def avg_sd(x): [avg(float(x)), stdev(float(x))];
4
5 beads = select * from good_opp_vct where pop = "beads";
6 bead_stats = select avg_sd(fsc_small) as [fsc_avg, fsc_sd],
7                  avg_sd(chl_small) as [chl_avg, chl_sd],
8                  avg_sd(pe) as [pe_avg, pe_sd],
9                  Cruise from beads;
10
11 store(bead_stats,
        armbrustlab:seaflo:bead_stats_v4_bycruise_untrans);
```

Query Language: MyriaL

Developer Options

- Profile Query

Profiling will make the query run a little bit slower but allows you to examine exactly...

Co...

Comp...

http://myria.cs.washington.edu

16

Analysis through Declarative Queries

Grab my data from Amazon S3

```
smallTable = load("https://s3-us-west-2.amazonaws.com/uwdb/sampleData/smallTable",  
csv(schema(column0:int, column1:int), skip=0));
```

...

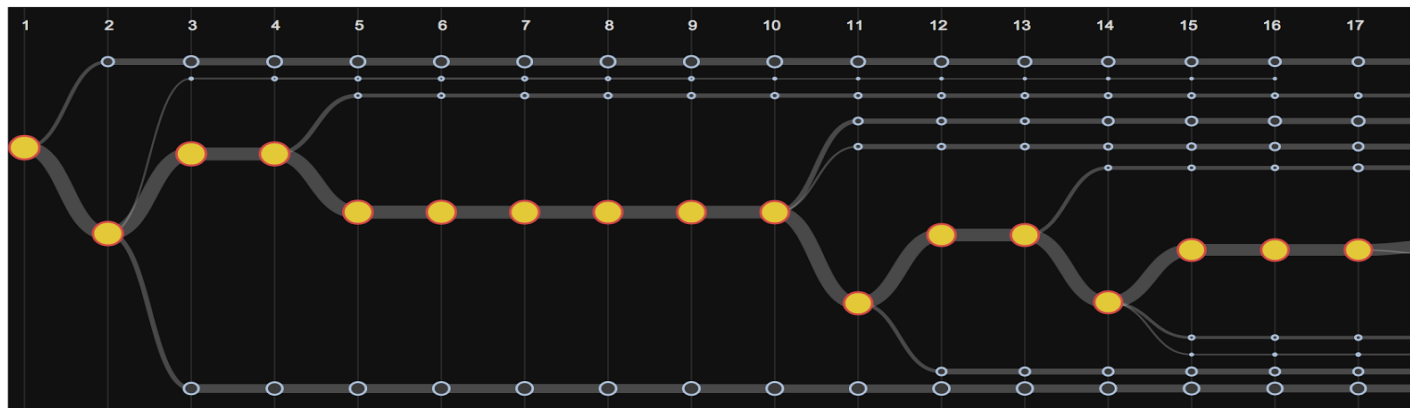
```
store(smallTable, smallTable, [$0]);
```

Perform some data manipulation

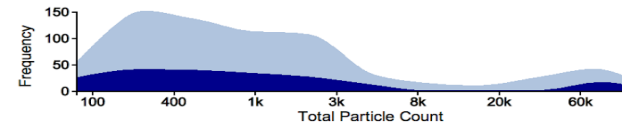
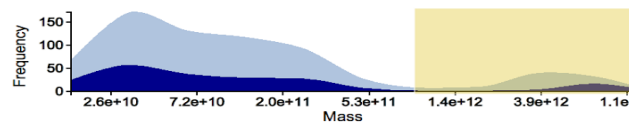
```
t = scan(public:adhoc:smallTable);  
smallTableAggregated = select count(*) from t;  
store(smallTableAggregated, smallTableAggregated);
```

Example: MyMergerTree

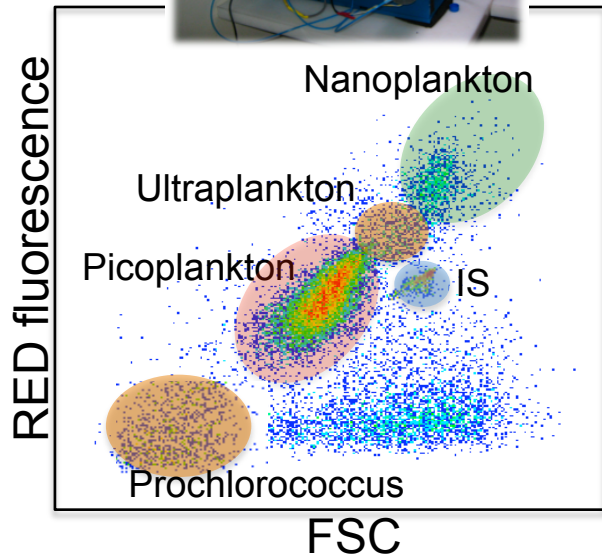
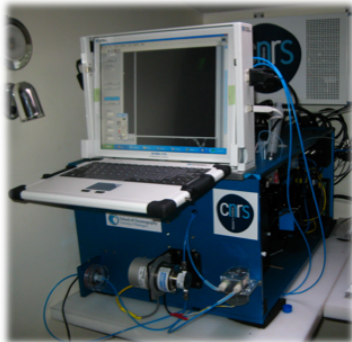
Cloud Service for analyzing galactic merger trees
Built on top of Myria
Used to analyze a 5TB dataset



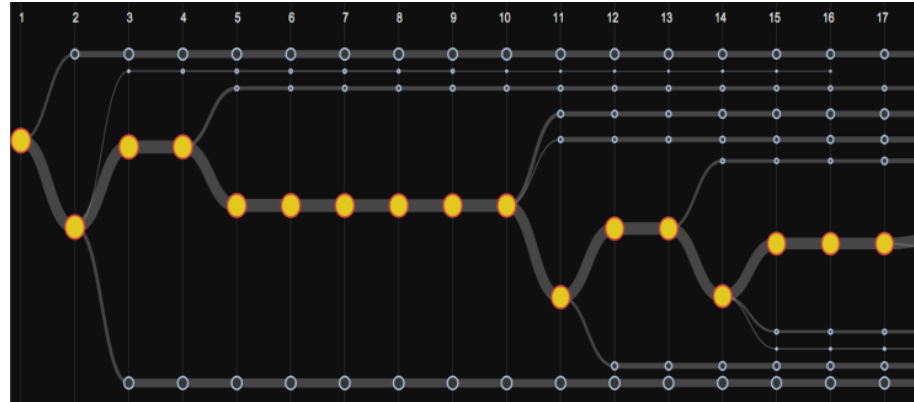
Highlights



Example Myria Applications



Environmental
Flow Cytometry

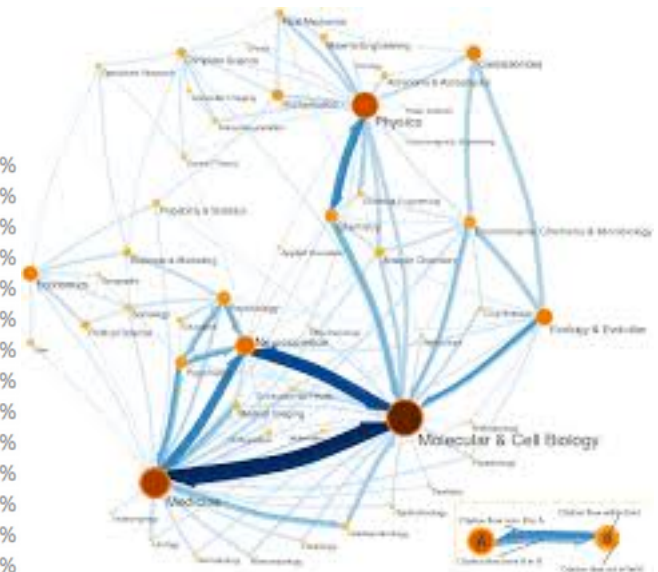


Galaxy Simulations

FREQUENT VALUES

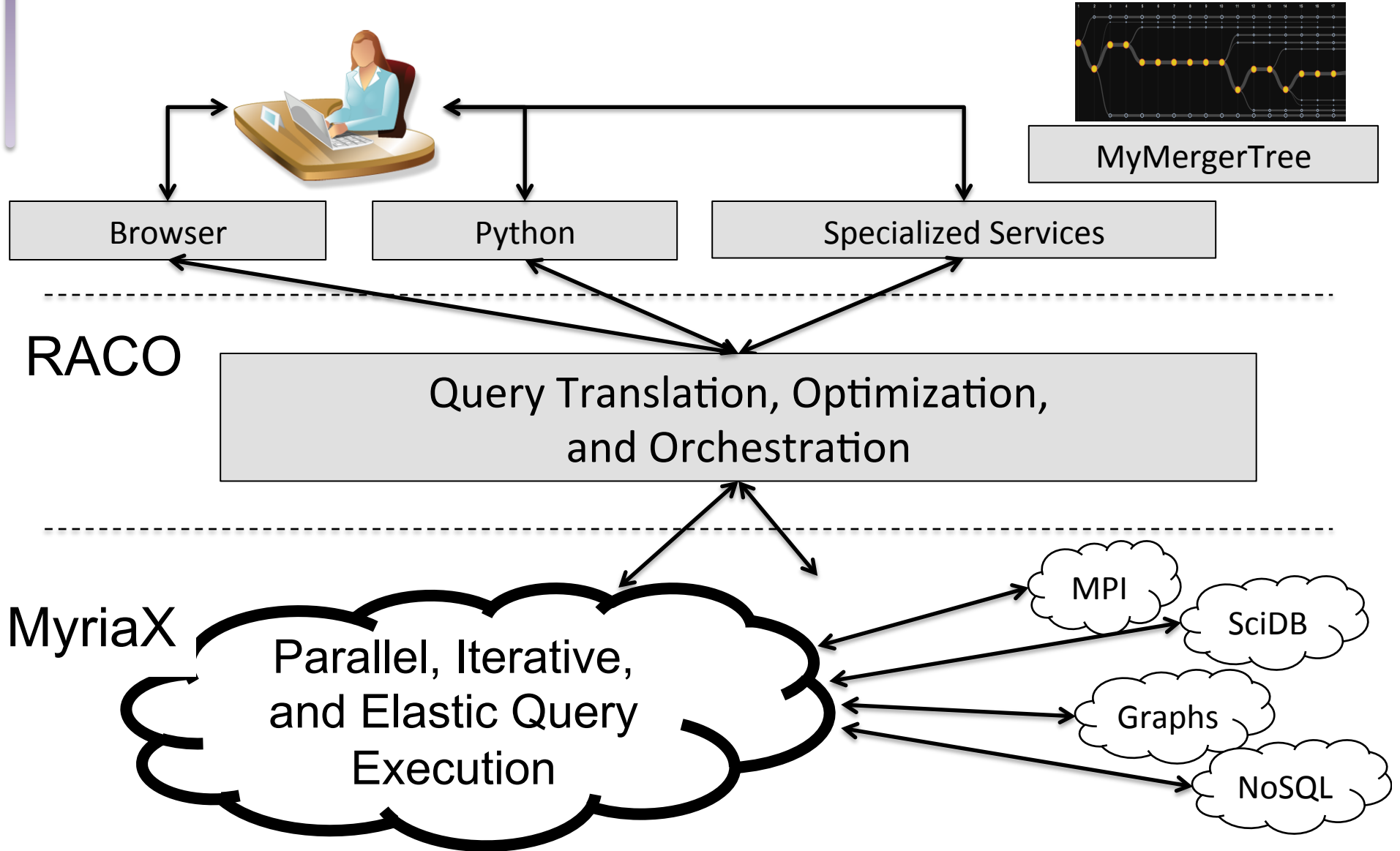
26532	3,282	0.8977%
2237	2,540	0.6947%
26530	2,383	0.6518%
0380	2,325	0.6359%
21589	1,759	0.4811%
0380-5	1,421	0.3887%
70921	1,400	0.3829%
21856	1,371	0.3750%
1134	1,232	0.3370%
1642	1,197	0.3274%
23039	1,189	0.3252%
21290	1,099	0.3006%
21193	1,027	0.2809%
6119	1,026	0.2806%
1298	1,021	0.2793%

Retail Analytics



Bibliometrics

Myria Is a Cloud Service



Myria is Expressive

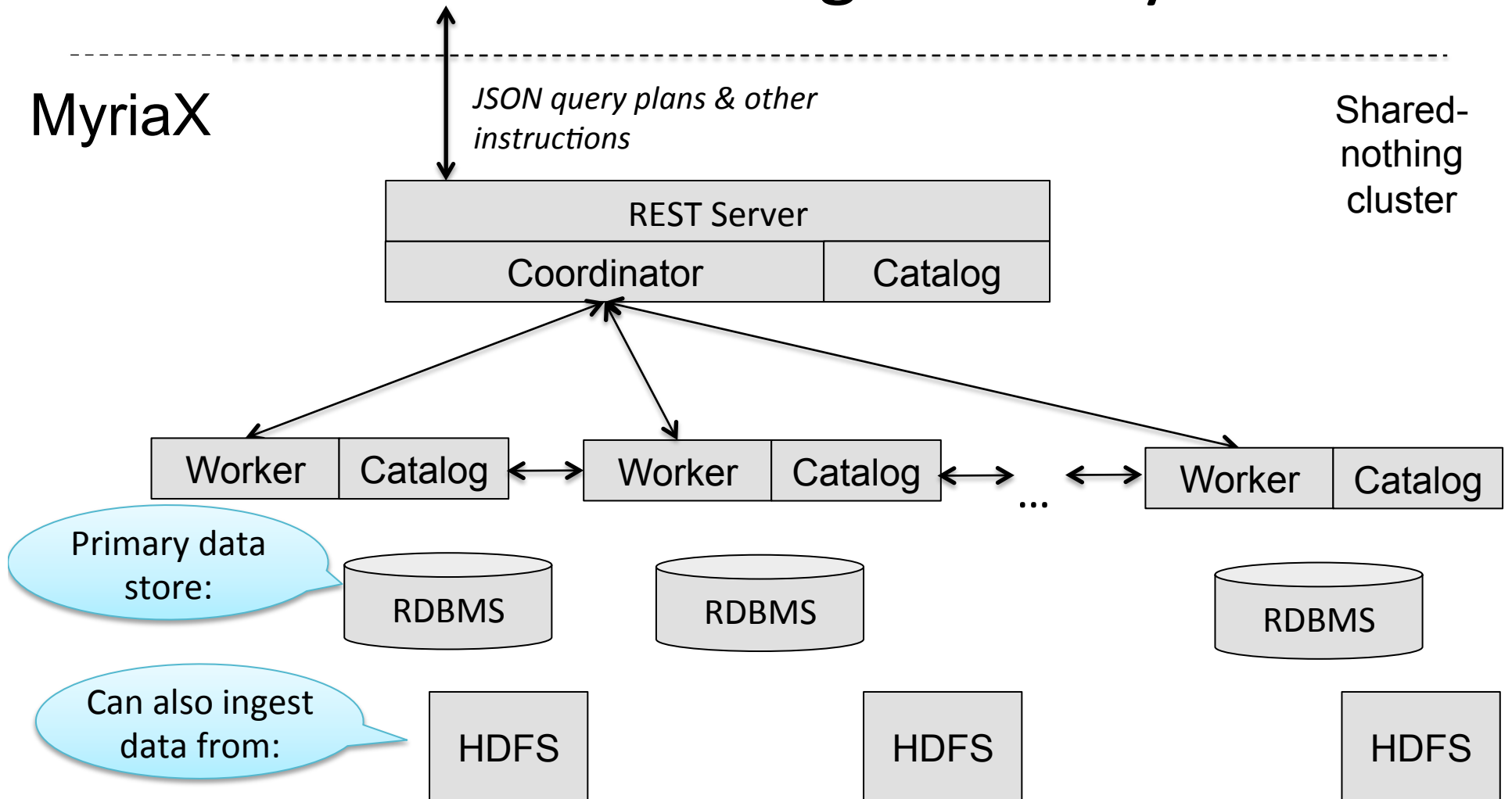
Myria is expressive

- SQL, Datalog, MyriaL

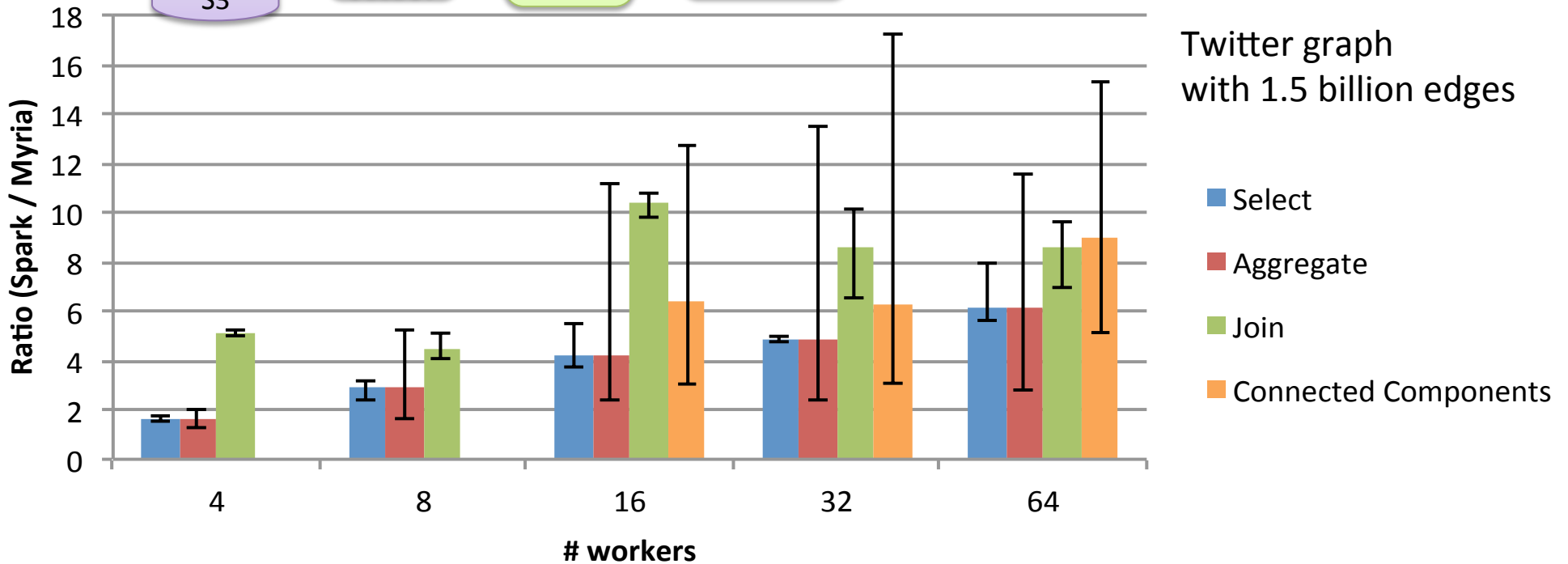
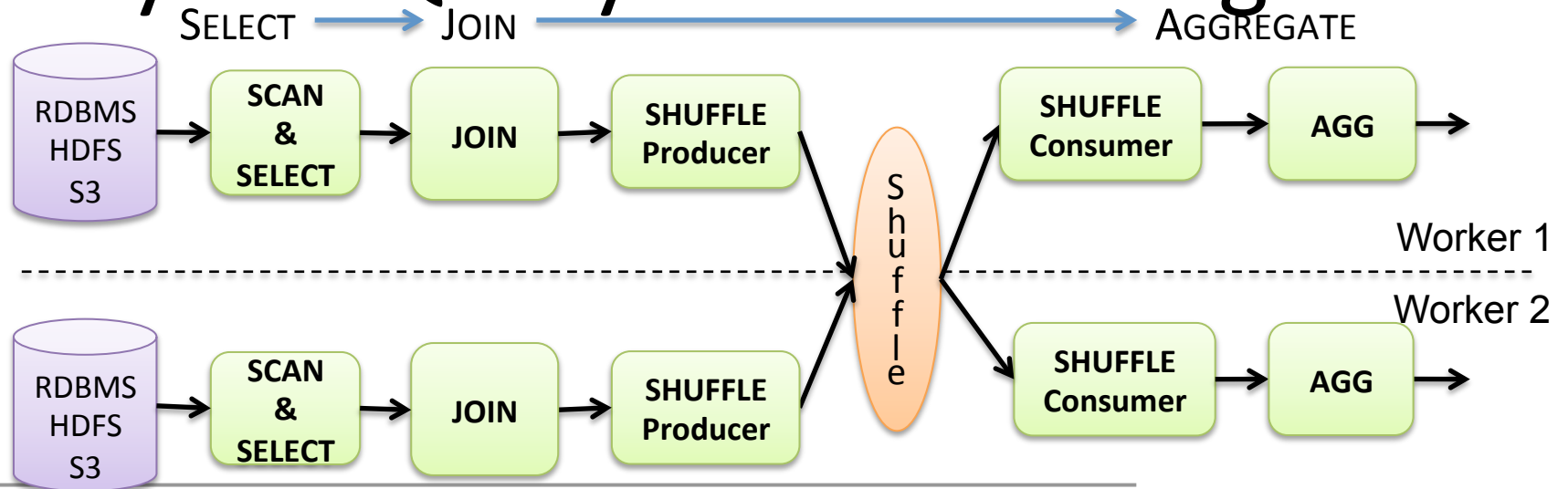
Quick example (connected components)

```
edgesRaw = scan(astro:cosmo50:cosmo50CompleteEdges);
edgesAll = [from edgesRaw as E where E.NowGroup = 52
            emit E.CurrentTime, E.CurrentHalo, E.NextHalo, E.SharedParticlesCount];
edgesNew = [from edgesAll as E where E.CurrentTime == 1
            emit E.CurrentTime, E.CurrentHalo, E.NextHalo, E.SharedParticlesCount];
edgesResult = edgesNew;
I = [1 as i];
do
  delta = [from edgesNew as E1, edgesAll as E2, I
           where E1.NextHalo == E2.CurrentHalo and E1.CurrentTime == I.i and E2.CurrentTime == I.i+1
           emit E2.CurrentTime as CurrentTime, E2.CurrentHalo as CurrentHalo, E2.NextHalo as NextHalo,
           E2.SharedParticlesCount as SharedParticlesCount];
  edgesResult = unionall(delta, edgesResult);
  edgesNew = delta;
  I = [from I emit i+1 as i];
while [from I emit min(i) <= 10];
store(edgesResult, finalAnswer);
```

Myria is a Parallel Data Management System



MyriaX Query Execution Engine



Example Specific Research Problems

Recursive Relational Query with Aggregations

- Positive Datalog
- Multiple relations with recursive dependencies

```
Edges (time, gid1, gid2, :- Galaxies (time, gid1),  
    $COUNT(*) as c)      Particles (time, gid1, pid),  
                           Particles (time+1, gid2, pid)  
Galaxies (time+1, gid2) :- Galaxies (time, gid1),  
                           Edges (time, gid1, gid2, c), c >= 2
```

- Aggregate functions:

- Only in head

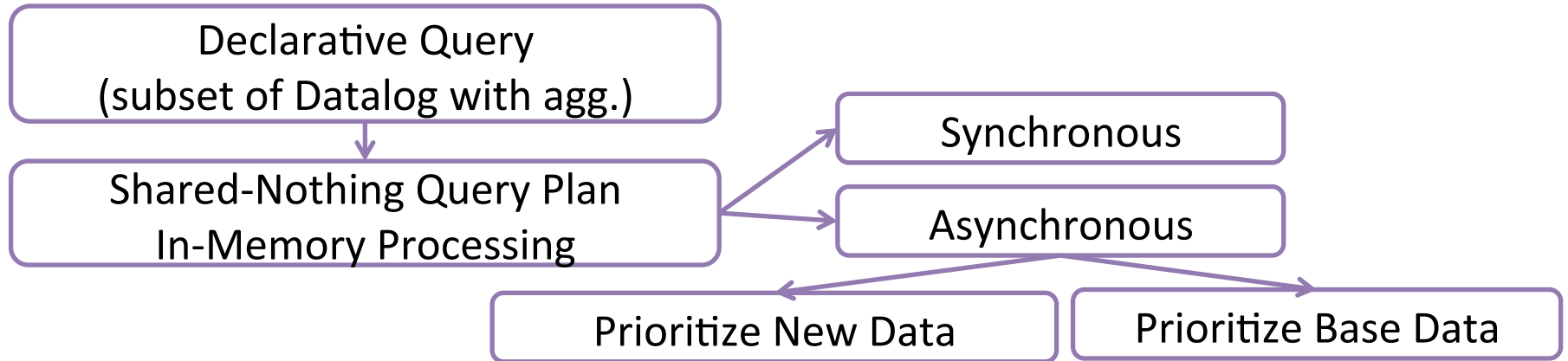
```
Edges (time, gid1, gid2, $COUNT(*) as c)
```

- Bag-monotonic

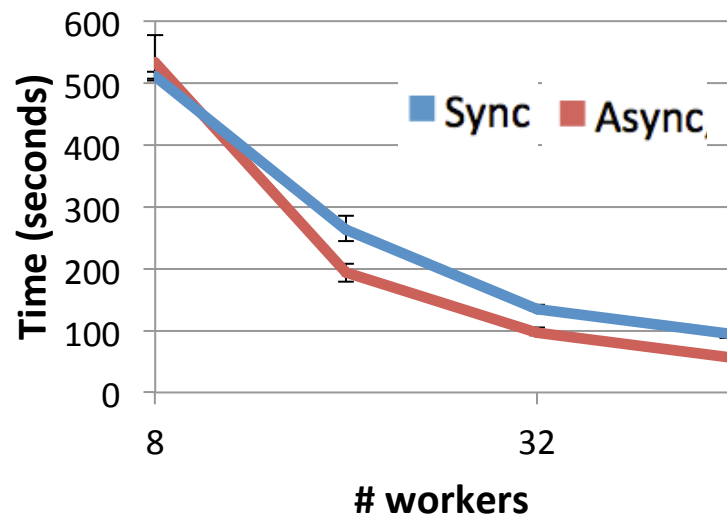
```
Edges (time, gid1, gid2, c), c >= 2
```

- A subset of Datalog-with-aggregation
 - Details in the paper

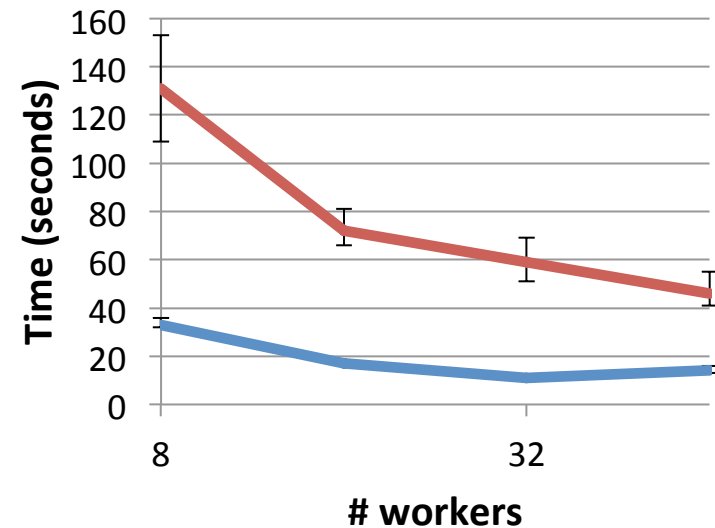
Important Runtime Optimizations



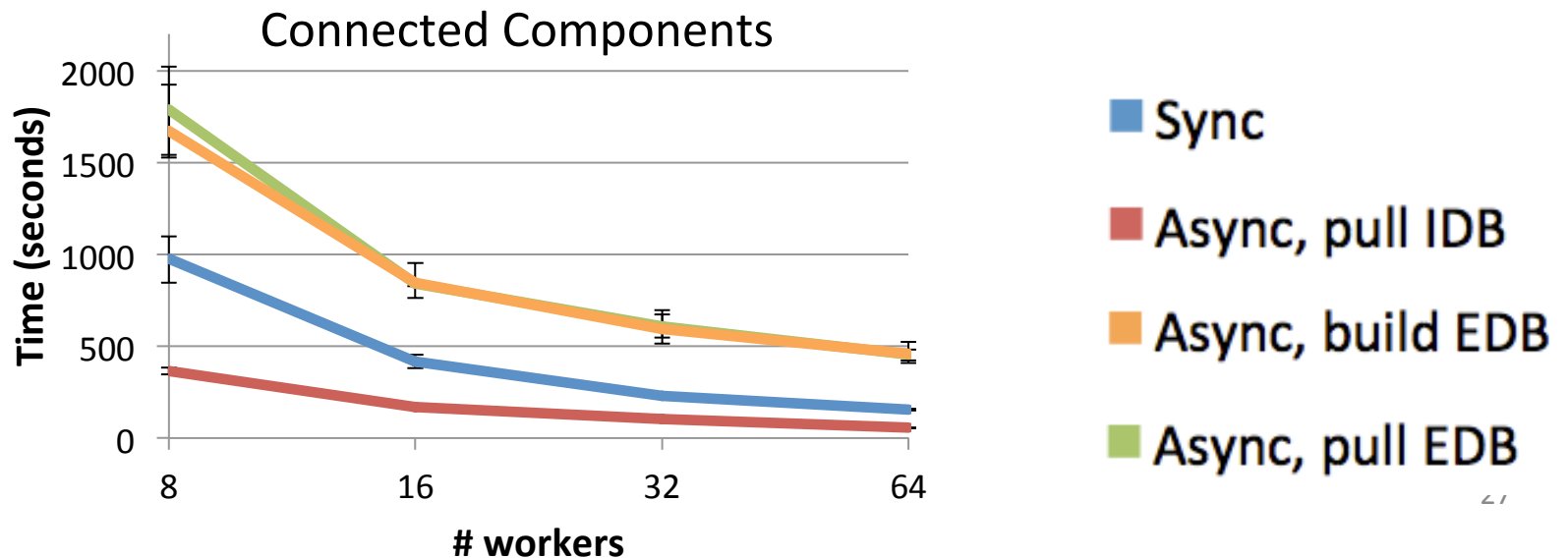
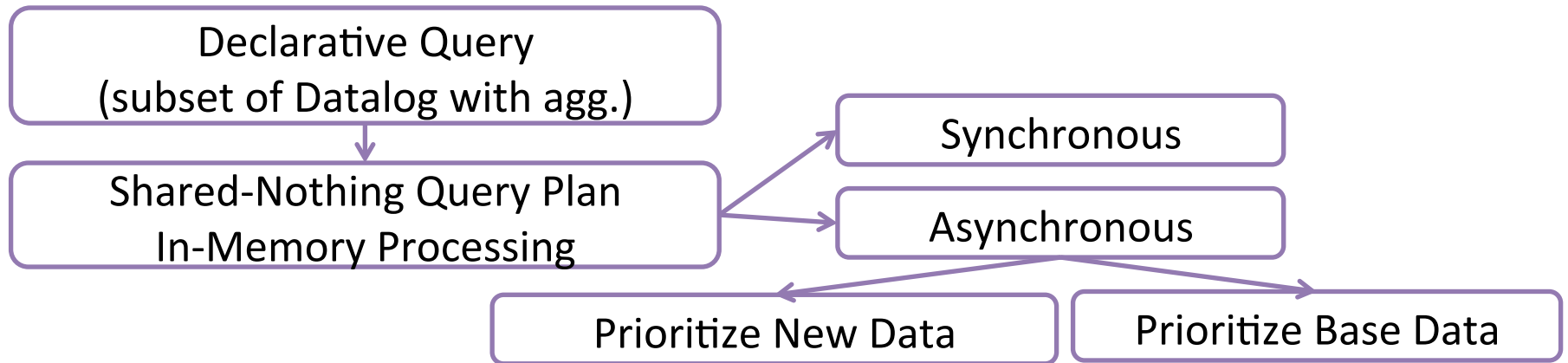
Galaxy evolution



Least Common Ancestor

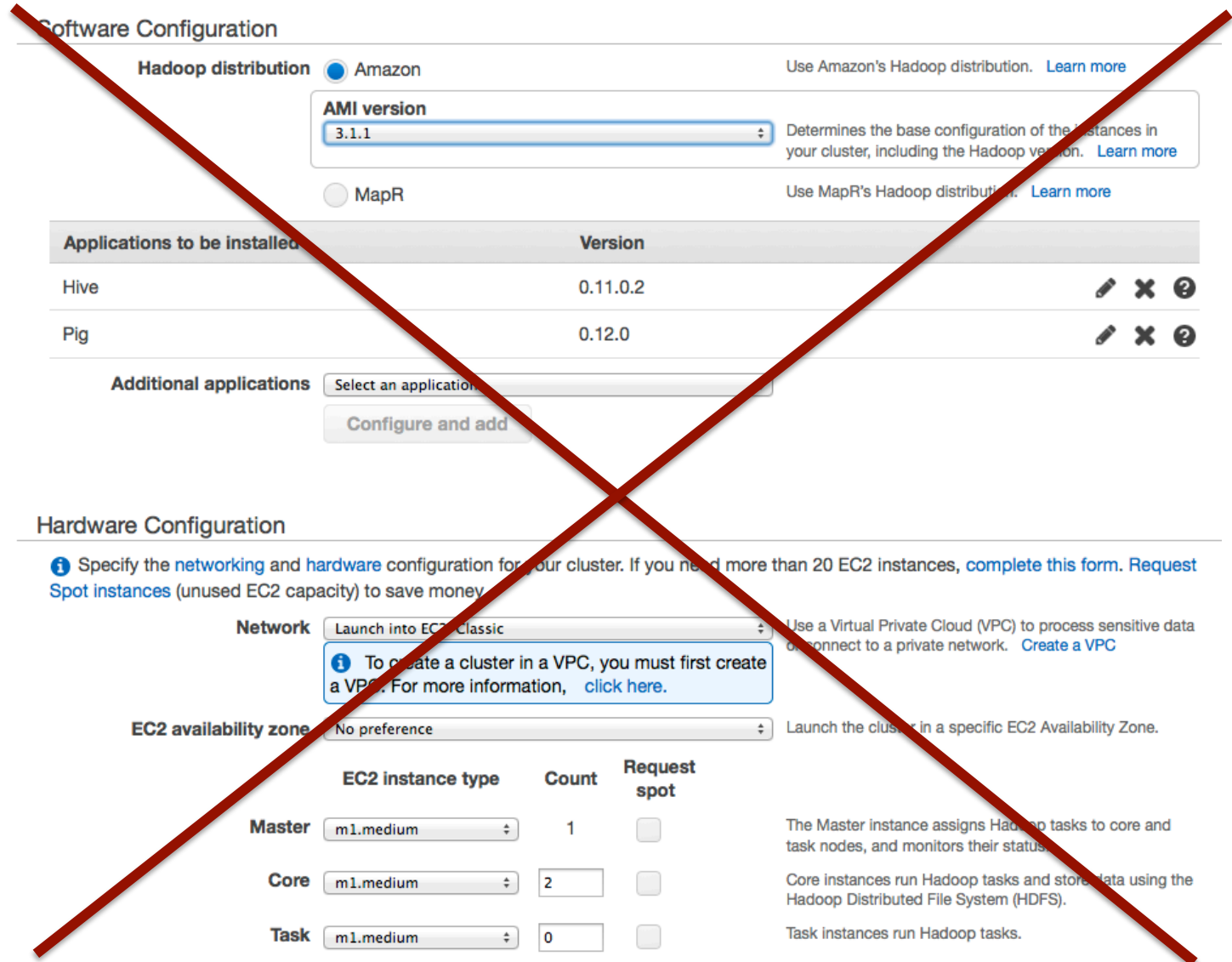


Important Runtime Optimizations



Myria Changes the Face of Cloud Services

Today:



Myria Changes the Face of Cloud Services

With Myria: Personalized Service Level Agreements

Available SLAs

Tier #1	
Query Template	Runtime (seconds)
SELECT (22 ATTR.) FROM (JOIN 4 Tables) WHERE '<10% CONDITION'	60
SELECT (22 ATTR.) FROM (JOIN 5 Tables) WHERE '<1% CONDITION'	
SELECT (27 ATTR.) FROM (JOIN 5 Tables) WHERE '<0.1% CONDITION'	
SELECT (9 ATTR.) FROM (JOIN 4 Tables) SELECT (27 ATTR.) FROM (JOIN 5 Tables) WHERE '<10% CONDITION'	120
SELECT (27 ATTR.) FROM (JOIN 5 Tables)	180
Purchase @ \$0.35/hour	

Tier #2, benefits in addition to Tier#1	
Query Template	Runtime (seconds)
SELECT (27 ATTR.) FROM (JOIN 5 Tables) WHERE '<10% CONDITION'	60
SELECT (27 ATTR.) FROM (JOIN 5 Tables)	120
Purchase @ \$0.89/hour	

Conclusion

- We live in the “big data era”
- We need to build new data management and analytics systems to handle the data
- We work with domain scientists and companies to use these tools and enable new discoveries and new services

Acknowledgments

- The University of Washington (UW) Database Group
 - UW eScience
 - UW astronomy
 - And many others
-
- Our sponsors: NSF, ISTC Big Data, HP Labs, Microsoft Research, Yahoo!, Google, Intel, Amazon, Facebook, UW eScience