Making Sense at Scale with Algorithms, Machines & People

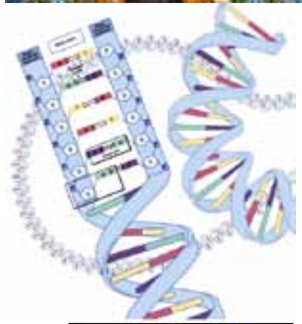
Tim Kraska
EECS, Computer Science
UC Berkeley

Agenda

- Issues/Opportunities of Big Data
- AMPLab Background
- AMP Technologies
  - Algorithms for Machine Learning
  - Machines for Cloud Computing
  - People for Crowd Sourcing
- Project Status and Wrap Up

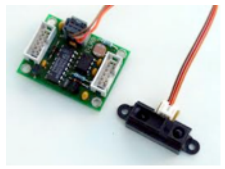
Big Data is Massive…

- Facebook:
  - 130TB/day: user logs
  - 200-400TB/day: 83 million pictures
- Google: > 25 PB/day processed data
- Data generated by LHC: 1PB/sec
- Gene sequencing:
  - Sequence 1 human cell costs Illumina $1k
  - Whole genome: $100,000 → 1.5 GB – 30 TB (raw)
  - Sequence 1 cell for every infant by 2015?
  - 10 trillion cells / human body
- Total data created in 2010: 1 ZettaByte (1,000,000 PB)/year
  - ~60% increase every year



… Growing …

- More and more devices



- Connected people

- Cheaper and cheaper storage: +50%/year

amplab

… and Dirty

- Diverse
- Variety of sources
- Uncurated
- No schema
- Inconsistent semantics
- Inconsistent syntax

Queries have issues too

- Diverse
- Time-sensitive
- Opportunistic
- Exploratory
- Multi-hypotheses Pitfall

“Big Data”: Working Definition

When the normal application of current technology doesn’t enable users to obtain **timely** and **cost-effective** answers of sufficient **quality** to data-driven questions

A Necessary Synergy

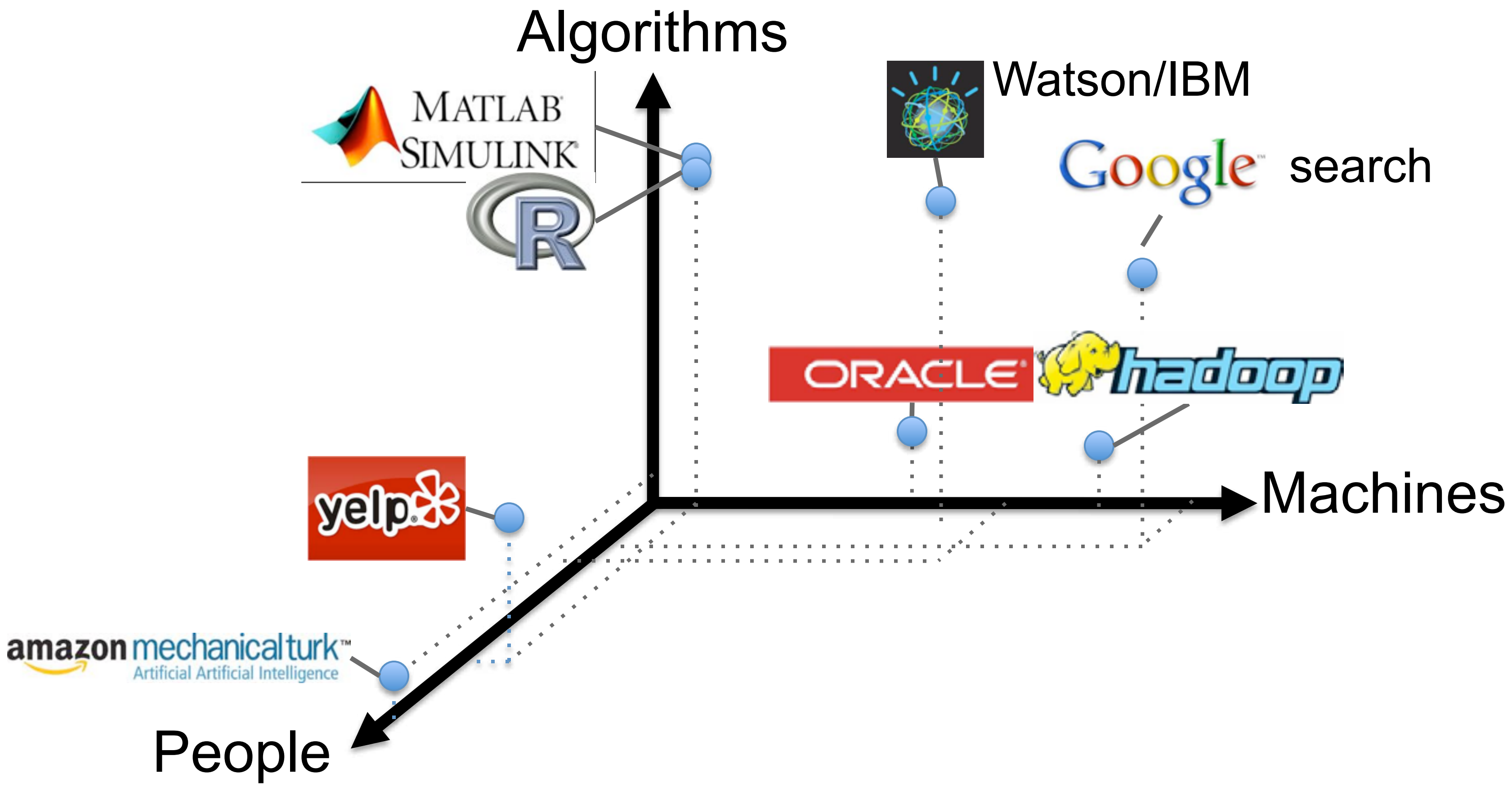
1. Improve scale, efficiency, and quality of machine learning and analytics to increase value from Big Data(**Algorithms**)

2. Use cloud computing to get value from Big Data and enhance datacenter infrastructure to cut costs of Big Data management (**Machines**)

3. Leverage human activity and intelligence to obtain data and extract value from Big Data cases that are hard for algorithms (**People**)

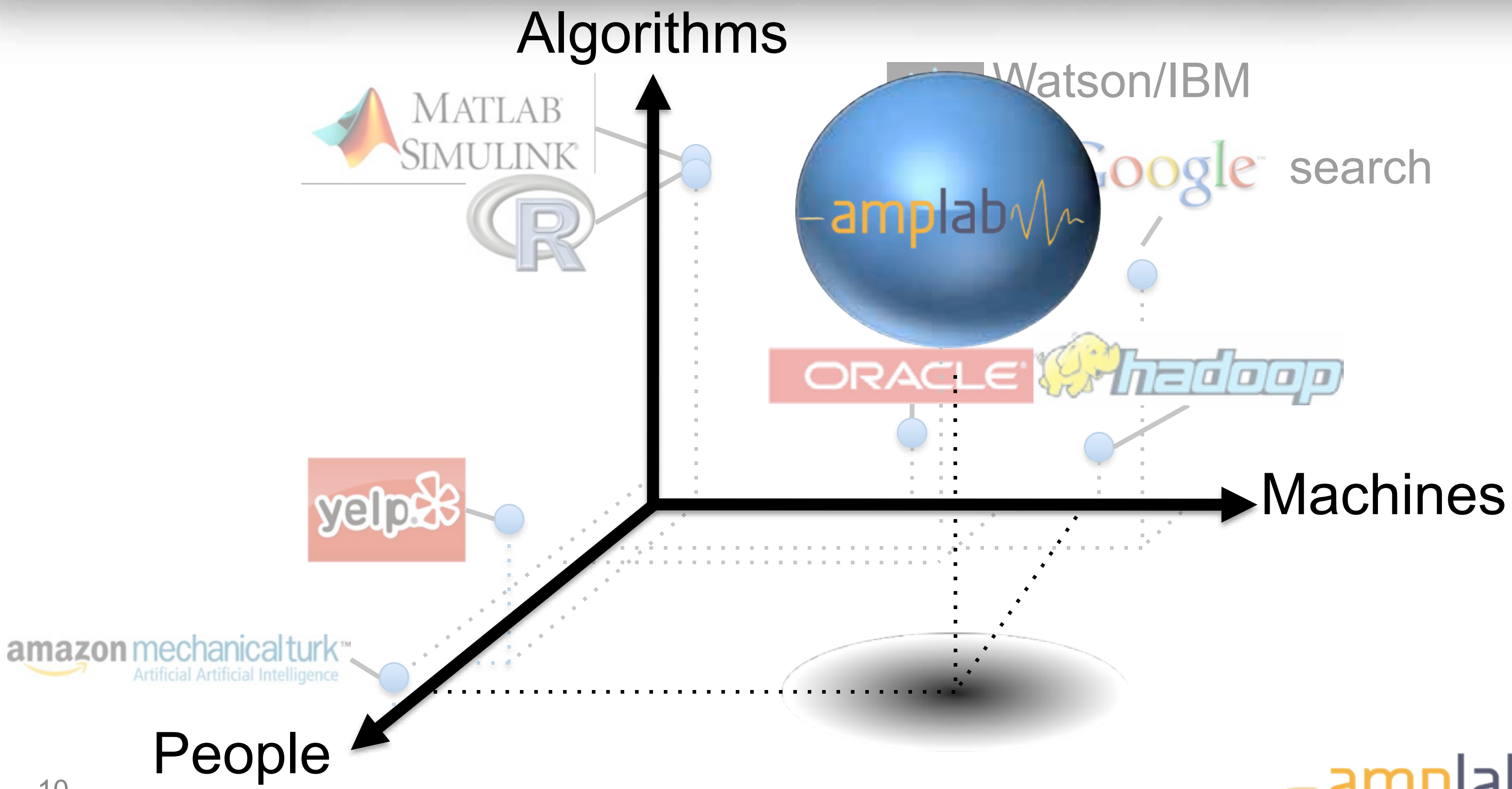
Algorithms, Machines, People

- Today's solutions:



The AMPLab

Making sense at scale by integrating Algorithms, Machines, and People



amplab

AMPLab: What is it?

A Five-Year research collaboration to develop a new generation of data analysis methods, tools, and infrastructure for making sense of data at scale

(Started Feb 2011)

AMP Faculty and Sponsors

Berkeley Faculty

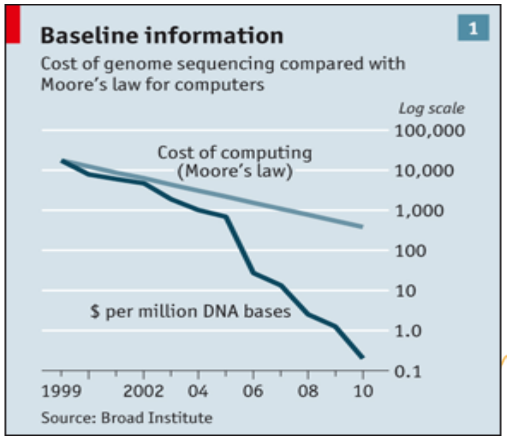
- Alex Bayen (mobile sensing platforms)
- Armando Fox (systems)
- Michael Franklin (databases) Director
- Michael Jordan (machine learning) Co-Director
- Anthony Joseph (security & privacy)
- Randy Katz (systems)
- David Patterson (systems)
- Ion Stoica (systems) Co-Director
- Scott Shenker (networking)

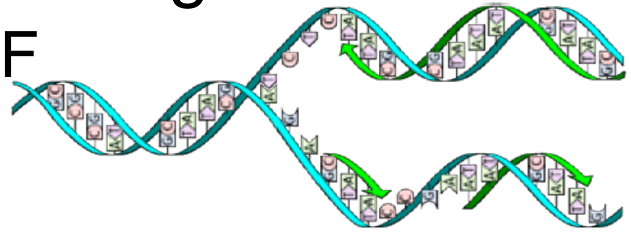
Sponsors: Google SAP

amazon.com web services, cloudera, ERICSSON, HUAWEI, IBM, intel, MarkLogic, Microsoft, NetApp, NEC, ORACLE, splunk

AMPLab Application Partners

- Mobile Millennium Project
  - *Alex Bayen*, Civil and Environment Engineering, UC Berkeley
- Microsimulation of urban development
  - *Paul Waddell*, College of Environment Design, UC Berkeley
- Crowd based opinion formation
  - *Ken Goldberg*, Industrial Engineering and Operations Research, UC Berkeley
- Personalized Sequencing
  - *Taylor Sittler*, UCSF



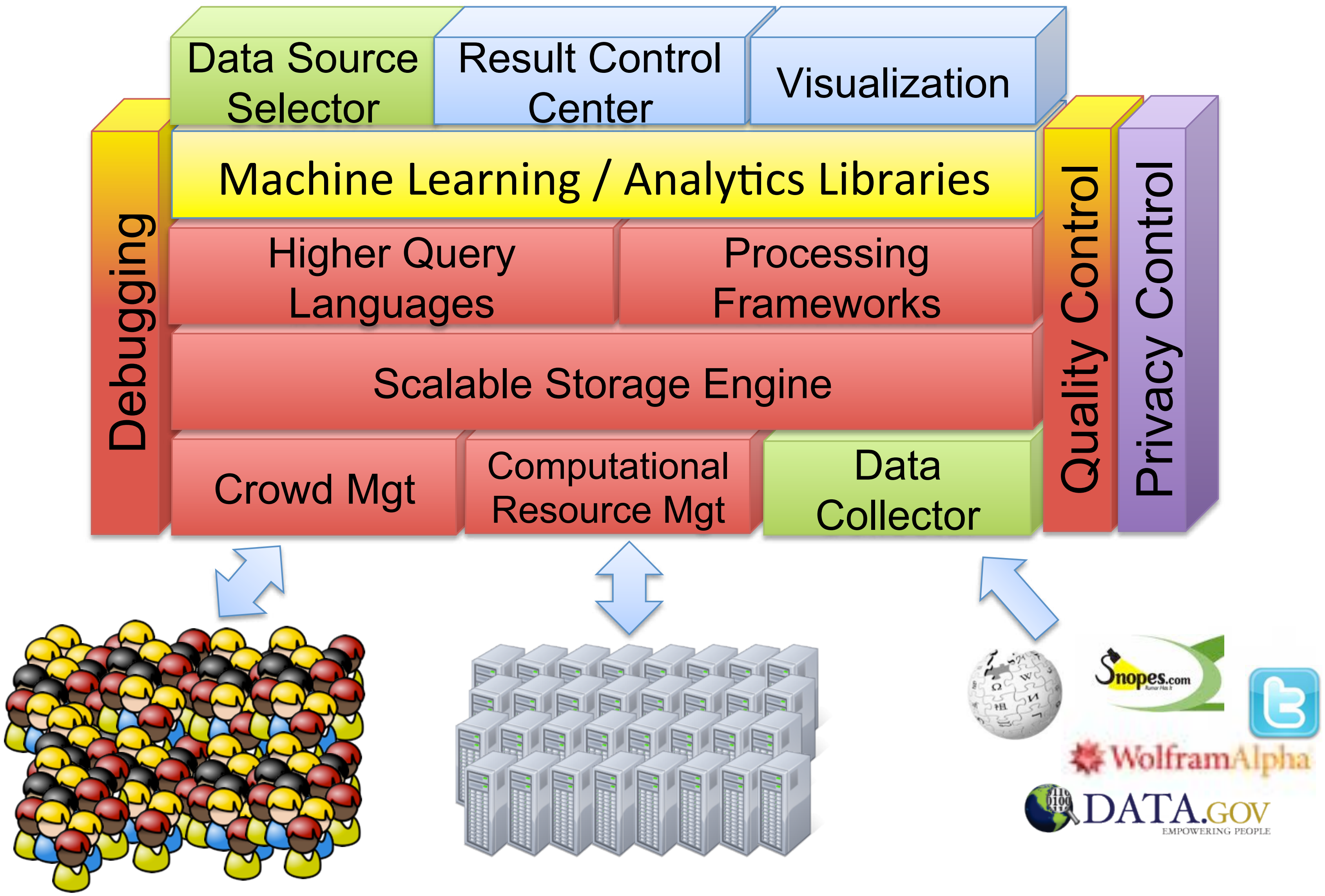


AMP Research Themes

- Algorithms
  - Scale up machine learning
  - Error bars on everything
- Machines
  - Datacenter as a computer
- People
  - People in all phases of data analysis
- AMP
  - Put it all together

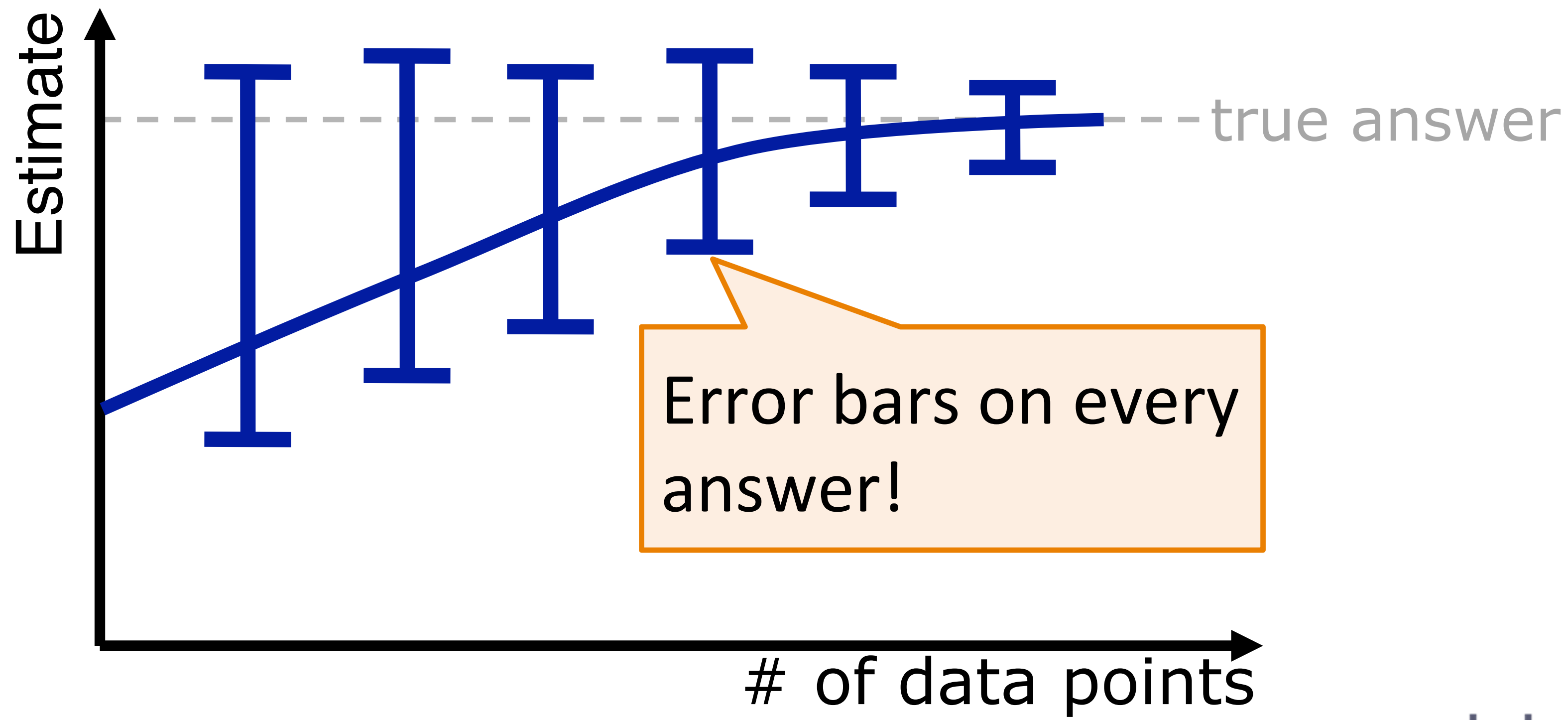
Berkeley Data Analytics System

Roles: 🟥 Infra. Builder 🟨 Algo/Tools 🟩 Data Collector 🟦 Data Analyst 🟪 Privacy Mgr



Algorithms: Error Management

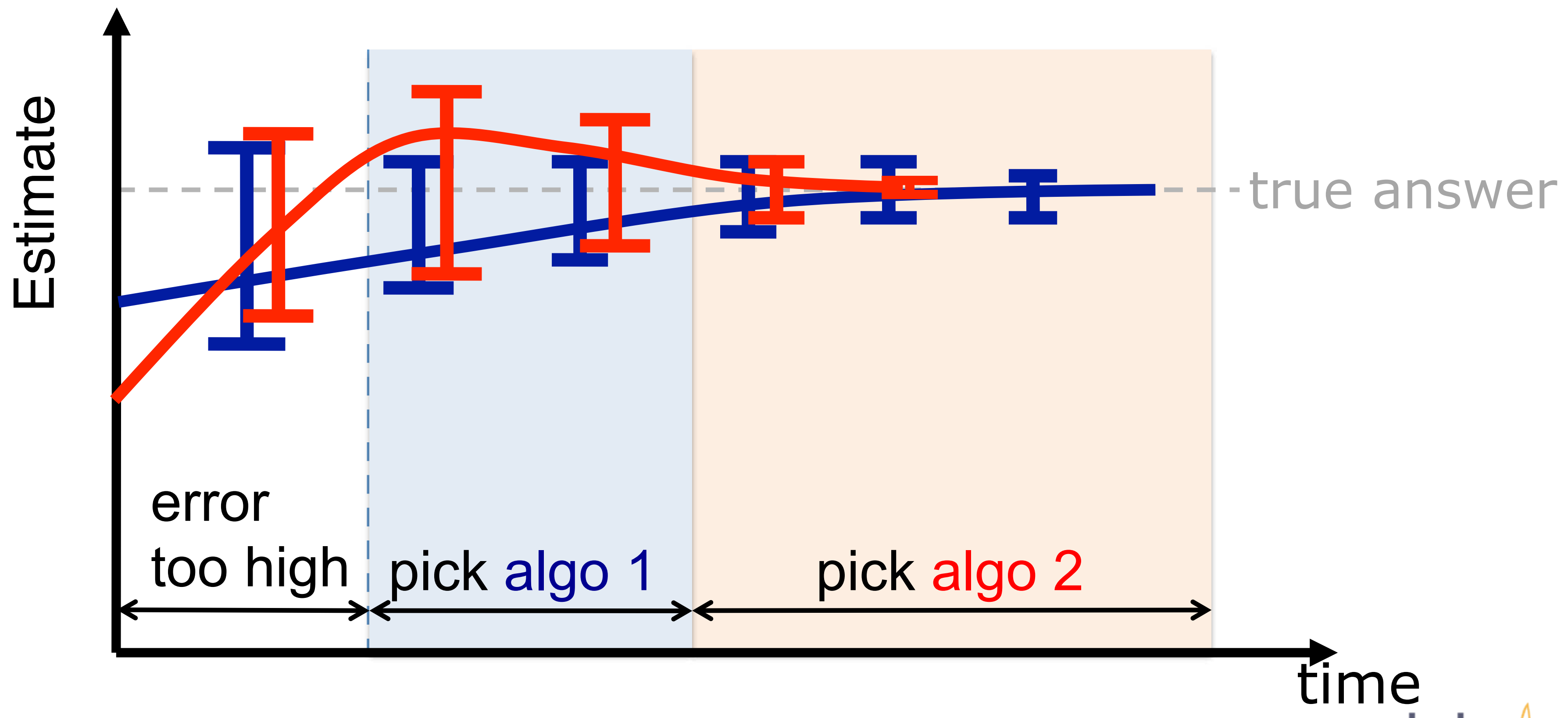
- Immediate results/continuous improvement
- Calibrate answer: provide error bars
- Breakthrough – "Big Data" Bootstrap



amplab

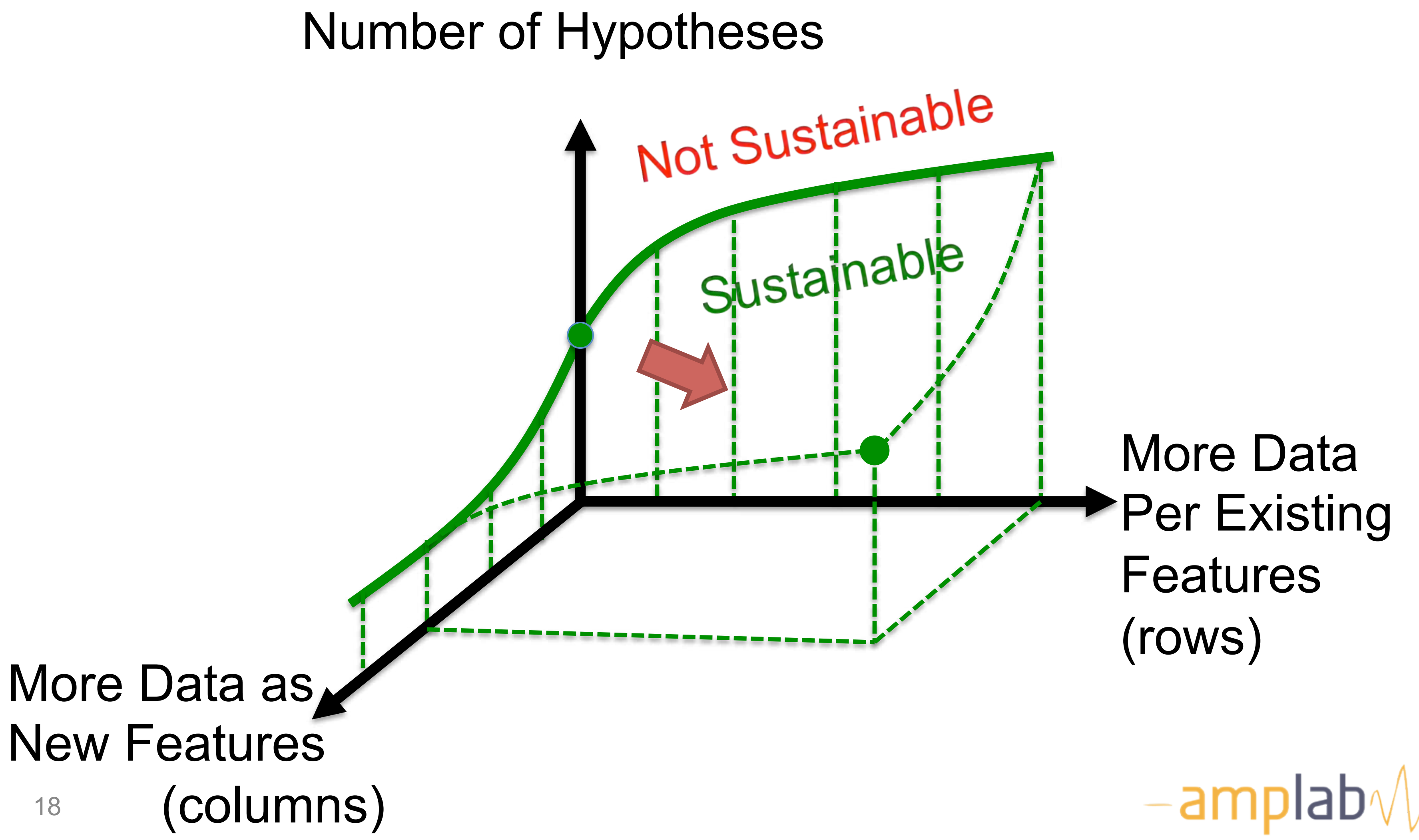
Algorithms: Error Management

- Given any problem, data and a time budget
  - Automatically pick the best algorithm
  - Actively learn and adapt strategy



Algos: Black Swans vs. Red Herrings

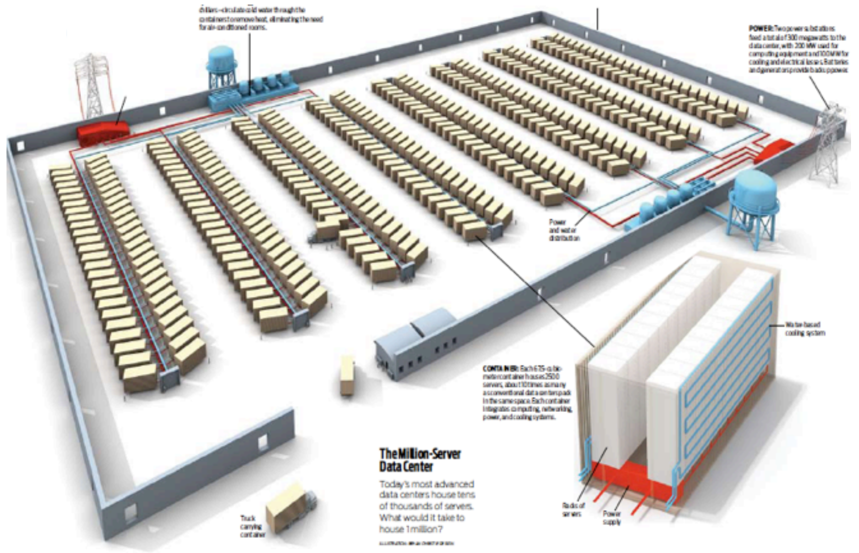
What about new data sources and bias?


Number of Hypotheses
Not Sustainable
Sustainable
More Data Per Existing Features (rows)
More Data as New Features (columns)

Machines

Goal: "The datacenter as a computer", but…

- – Special purpose clusters, e.g., Hadoop cluster
- – Highly variable performance
- – Hard to program
- – Hard to debug


Chillers—circulate cold water through the containers to remove heat, eliminating the need for air-conditioned rooms.
POWER: Two power substations feed a total of 300 megawatts to the data center, with 200 MW used for computing equipment and 100 MW for cooling and electrical losses. Batteries and generators provide backup power.
Power and water distribution
Water-based cooling system
CONTAINER: Each 67.5-cubic-meter container houses 2500 servers, about 10 times as many as conventional data centers pack in the same space. Each container integrates computing, networking, power, and cooling systems.
The Million-Server Data Center
Today's most advanced data centers house tens of thousands of servers. What would it take to house 1 million?
Truck carrying container
Racks of servers
Power supply

=?

amplab

Machines: Problem

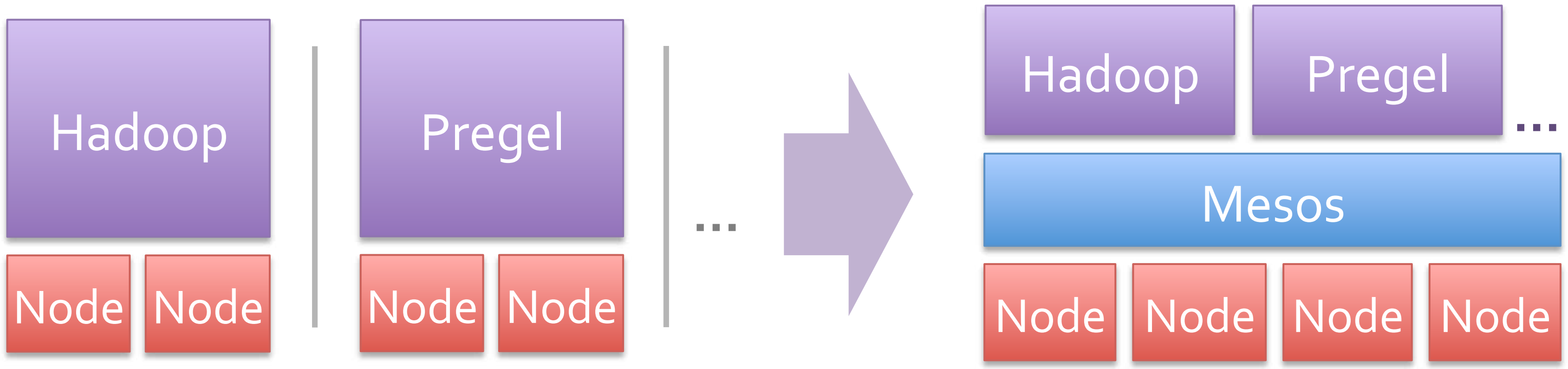
- Rapid innovation in cluster computing frameworks
  - Hadoop, Pregel, Dryad, Rails, Spark, Pig, MPI2, …
- No single framework optimal for all applications
- Want to run multiple frameworks in a single cluster

» …to *maximize utilization*

» …to *share data* between frameworks

Machines: A Solution

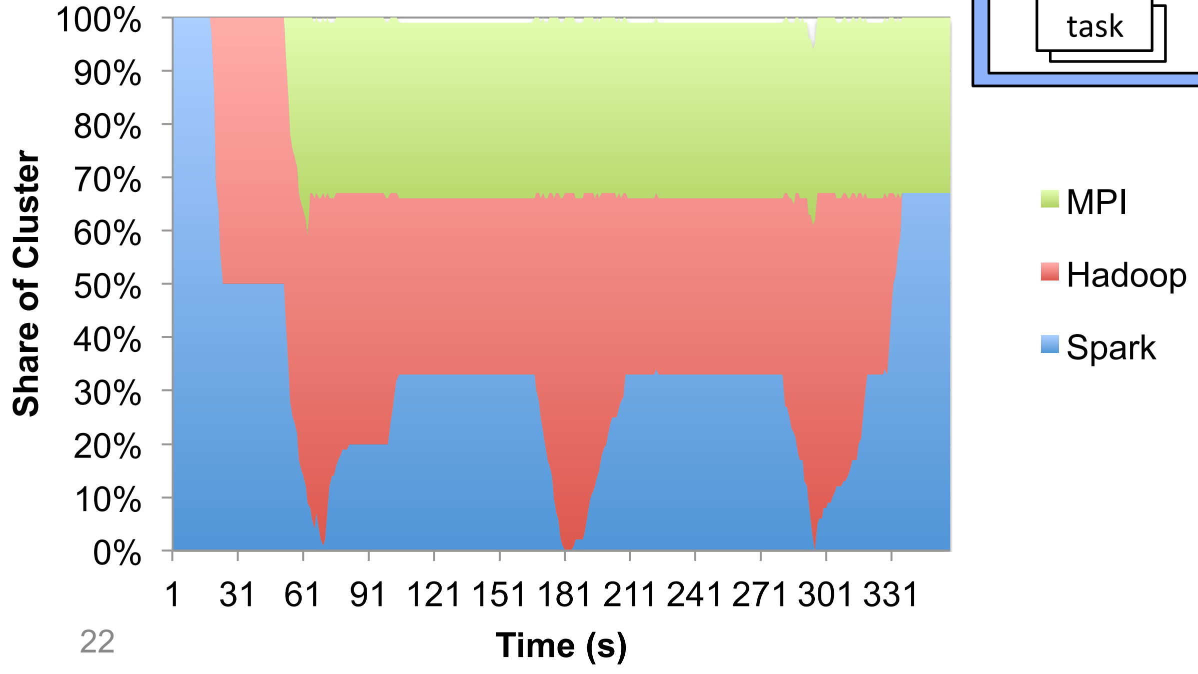
- Mesos: a resource sharing layer supporting diverse frameworks
  - – Fine-grained sharing: Improves utilization, latency, and data locality
  - – Resource offers: Simple, scalable application-controlled scheduling mechanism

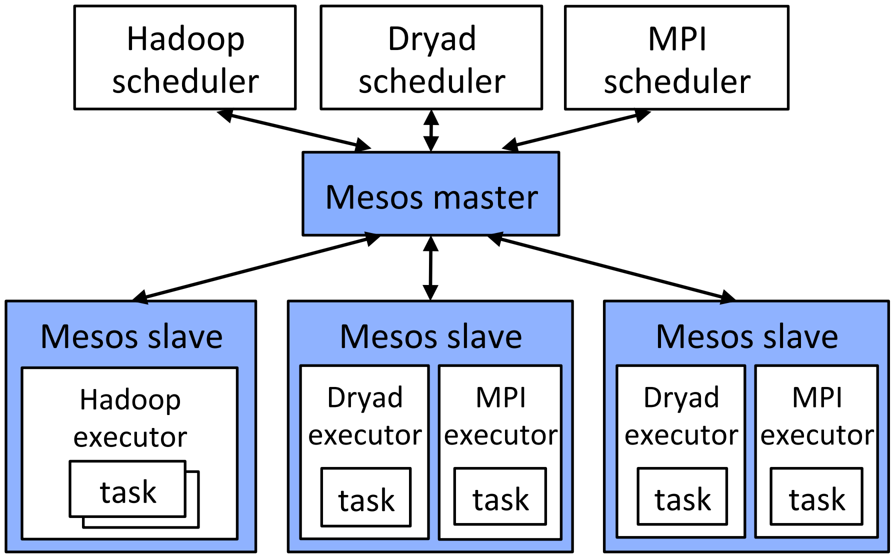


B. Hindman, et al, Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center, *NSDI 2011*, March 2011.

Mesos – Cluster Operating System

Efficiently shares resources among diverse parallel applications





Mesos: Implementation Status

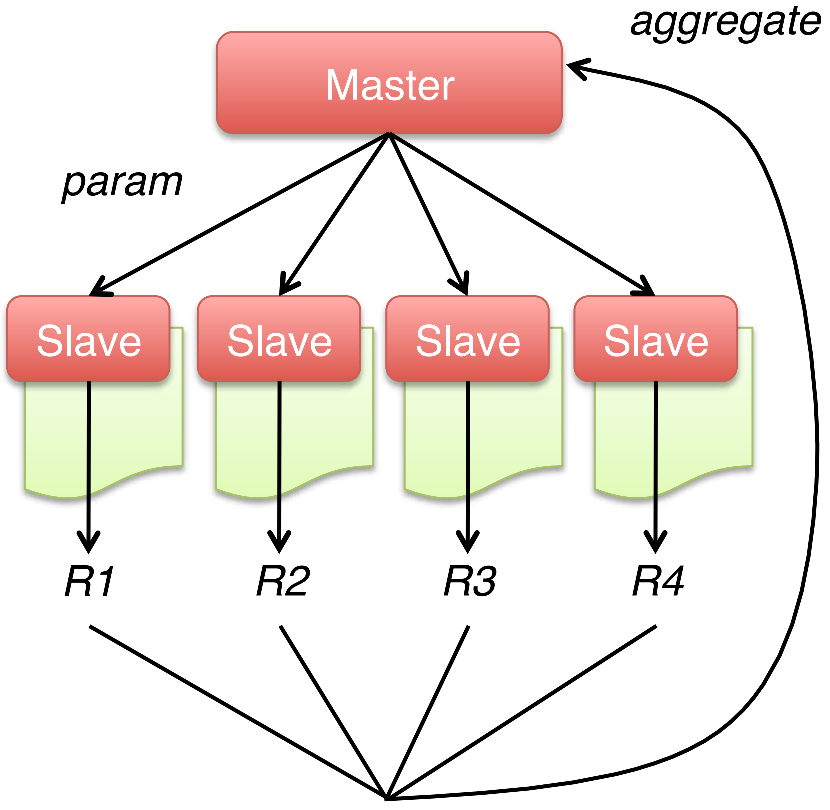
- 20,000 lines of C++
- Master failover using ZooKeeper
- Frameworks ported: Hadoop, MPI, Torque
- New specialized frameworks: Spark
- Open source in Apache Incubator
- In use at Berkeley, UCSF, Twitter and elsewhere

Datacenter Programming Framework: Spark

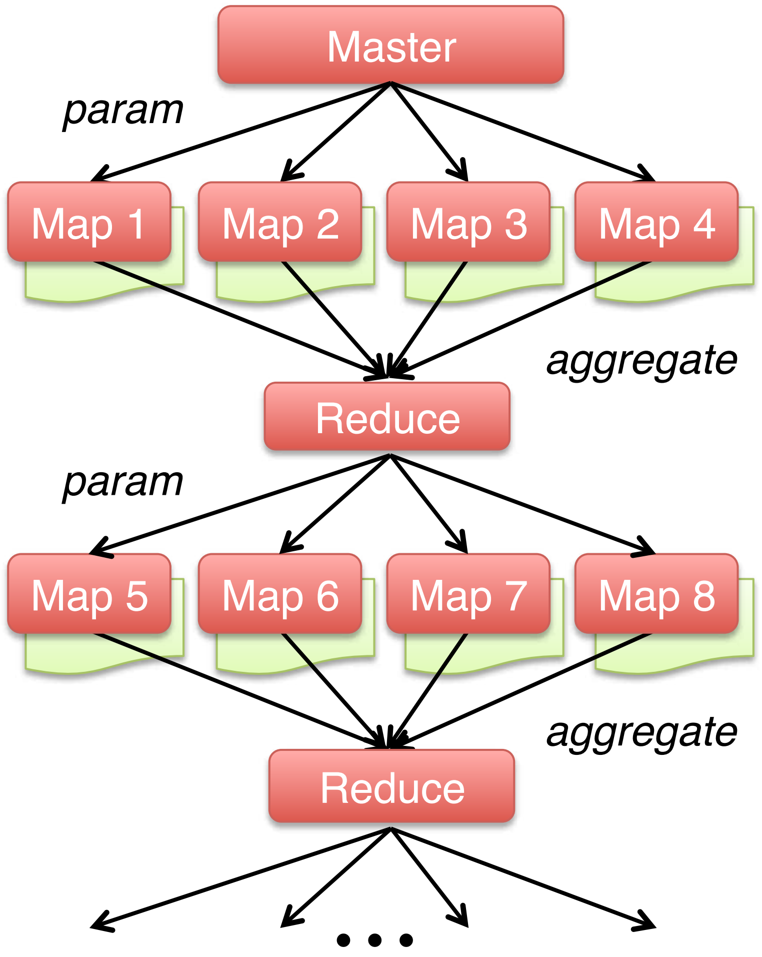
- In-memory cluster computing framework for applications that reuse working sets of data
  - *Iterative* algorithms: machine learning, graph processing, optimization
  - *Interactive* data mining: order of magnitude faster than disk-based tools
- Key idea: "resilient distributed datasets" that can automatically be rebuilt on failure
  - mechanism based on "lineage"

M. Zaharia, et al, Spark: Cluster Computing with Working Sets, *2nd USENIX Workshop on Hot Topics in Cloud Computing* (*Hot Cloud 2010*), June2010.

Spark Data Flow vs. Map Reduce



Spark



MapReduce / Dryad

amplab

Scala Language Integration

Serial Version

```
val data = readData(...)


var w = Vector.random(D)

for (i <- 1 to ITERATIONS) {
  var gradient = Vector.zeros(D)

  for (p <- data) {
    val s = (1/(1+exp(-p.y*
      (w dot p.x)))-1) * p.y
    gradient += s * p.x
  }
  w -= gradient
}

println("Final w: " + w)
```

Spark Version

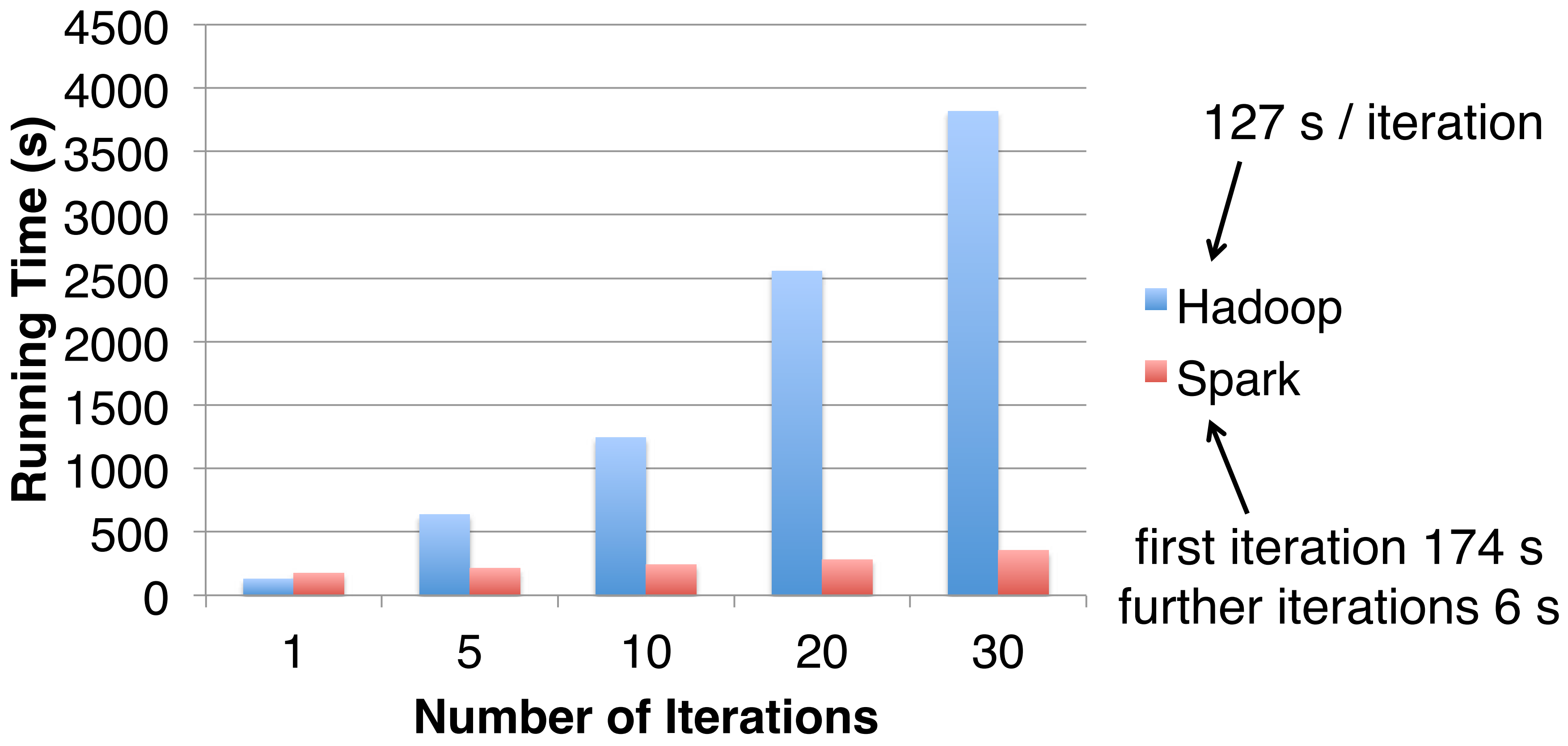
```
val data = spark.hdfsTextFile(...)
  .map(readPoint _).cache()

var w = Vector.random(D)

for (i <- 1 to ITERATIONS) {
  var gradient = spark.accumulator(
    Vector.zeros(D))
  for (p <- data) {
    val s = (1/(1+exp(-p.y*
      (w dot p.x)))-1) * p.y
    gradient += s * p.x
  }
  w -= gradient.value
}

println("Final w: " + w)
```

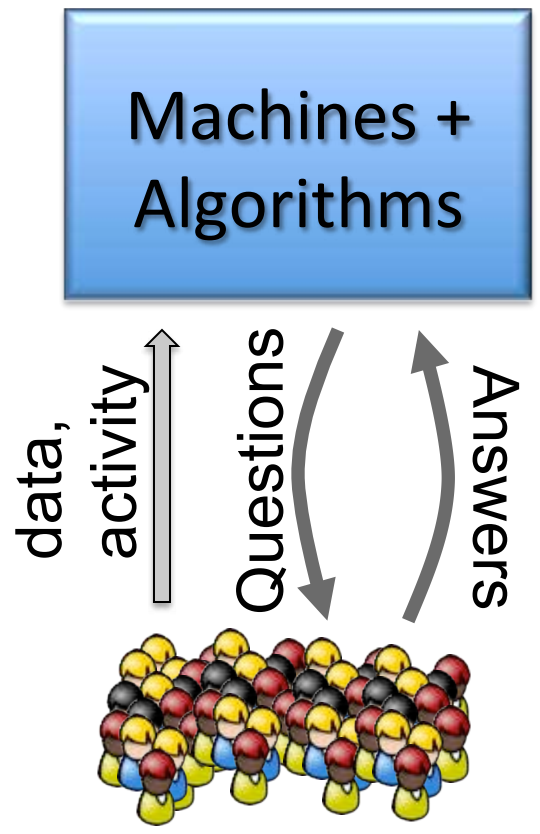
Logistic Regression Performance



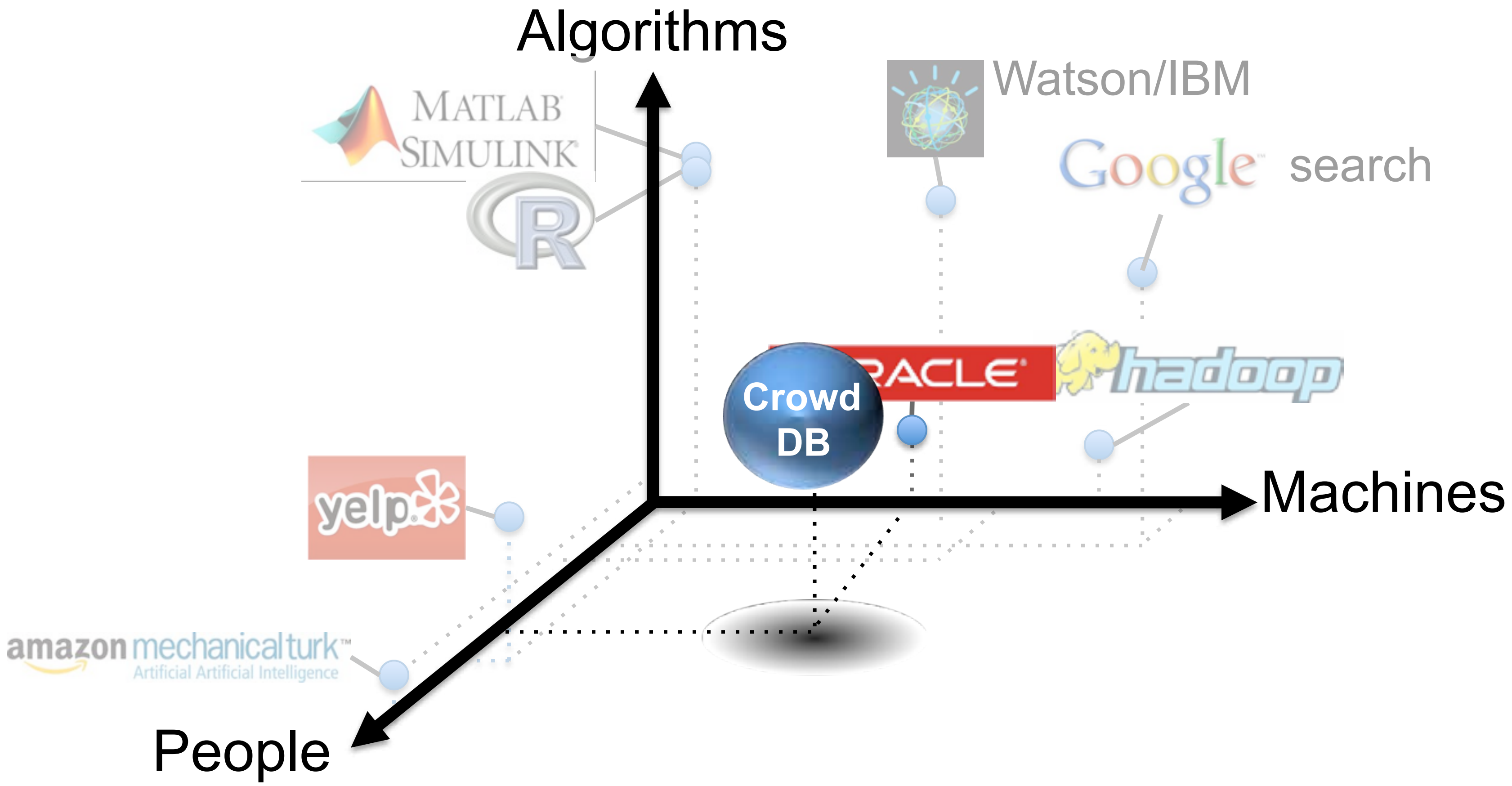
People

- Make people an integrated part of the system!
  - Leverage human activity
  - Leverage human intelligence (crowdsourcing):
    - Curate and clean dirty data
    - Answer imprecise questions
    - Test and improve algorithms

- Challenge
  - Inconsistent answer quality in all dimensions (e.g., type of question, time, cost)



CrowdDB – A First Step



amplab

Problem: DB-hard Queries

Company_Name	Address	Market Cap
Google	Googleplex, Mtn. View CA	$170Bn
Intl. Business Machines	Armonk, NY	$203Bn
Microsoft	Redmond, WA	$206Bn

```
SELECT Market_Cap
From Companies
Where Company_Name = “IBM”
```

```
Number of Rows: 0
```

Problem:
Entity Resolution

DB-hard Queries

Company_Name	Address	Market Cap
Google	Googleplex, Mtn. View CA	$170Bn
Intl. Business Machines	Armonk, NY	$203Bn
Microsoft	Redmond, WA	$206Bn

```
SELECT Market_Cap
From Companies
Where Company_Name = "Apple"
```

```
Number of Rows: 0
```

```
Problem:
Closed World Assumption
```

DB-hard Queries

```
SELECT Top_1 Image
From Pictures
Where Topic = “Business Success”
Order By Relevance
```

```
Number of Rows: 0
```

Problem:
Subjective Comparison

Easy Queries

Company_Name	Address	Market Cap
Google	Googleplex, Mtn. View CA	$170Bn
Intl. Business Machines	Armonk, NY	$203Bn
Microsoft	Redmond, WA	$206Bn

```
SELECT Market_Cap
From Companies
Where Company_Name = “IBM”
```

```
    $203Bn
    Number of Rows: 1
```

Pretty Easy Queries

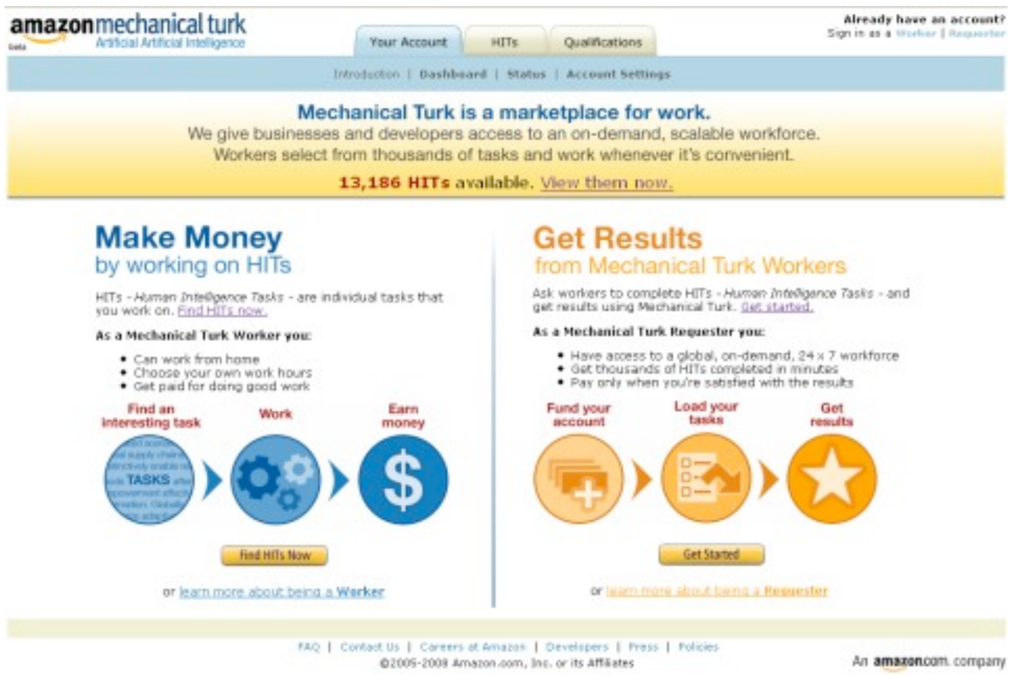
Company_Name	Address	Market Cap
Google	Googleplex, Mtn. View CA	$170Bn
Intl. Business Machines	Armonk, NY	$203Bn
Microsoft	Redmond, WA	$206Bn

```
SELECT Market_Cap
From Companies
Where Company_Name =
     “The Cool Software Company”
```

```
$xxxBn
Number of Rows: 1
```

Microtasking Marketplaces

- Current leader: Amazon Mechanical Turk
- Requestors place Human Intelligence Tasks (HITs)
  - Requestors approve jobs and payment
  - API-based: "createHit()", "getAssignments()", "approveAssignments()"
  - Other parameters: #of replicas, expiration, <u>**User Interface**</u>,…



amazon mechanical turk
Artificial Artificial Intelligence
beta

Your Account | HITs | Qualifications

Already have an account?
Sign in as a Worker | Requester

Introduction | Dashboard | Status | Account Settings

Mechanical Turk is a marketplace for work.
We give businesses and developers access to an on-demand, scalable workforce.
Workers select from thousands of tasks and work whenever it's convenient.
13,186 HITs available. View them now.

Make Money
by working on HITs

HITs - *Human Intelligence Tasks* - are individual tasks that you work on. Find HITs now.

As a Mechanical Turk Worker you:

- Can work from home
- Choose your own work hours
- Get paid for doing good work

or learn more about being a **Worker**

Get Results
from Mechanical Turk Workers

Ask workers to complete HITs - *Human Intelligence Tasks* - and get results using Mechanical Turk. Get started.

As a Mechanical Turk Requester you:

- Have access to a global, on-demand, 24 x 7 workforce
- Get thousands of HITs completed in minutes
- Pay only when you're satisfied with the results

or learn more about being a **Requester**

FAQ | Contact Us | Careers at Amazon | Developers | Press | Policies
©2005-2008 Amazon.com, Inc. or its Affiliates

An amazon.com company

- Workers (a.k.a. "turkers") choose jobs, do them, get paid

amplab

Idea: CrowdDB

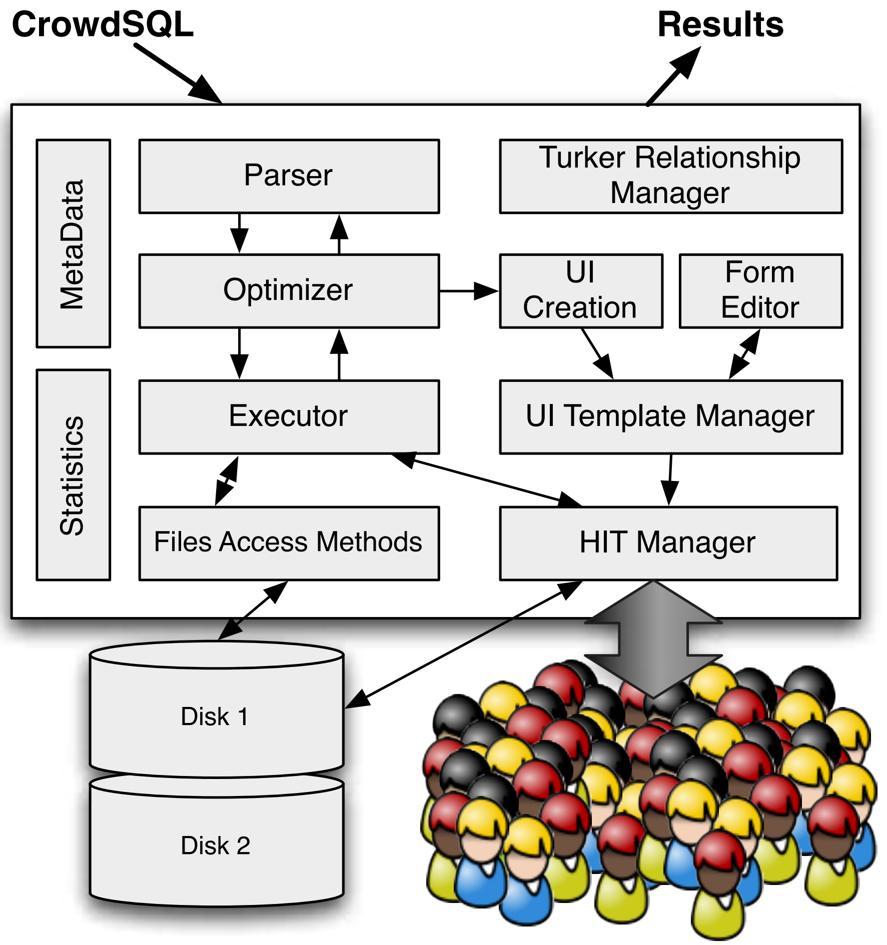
Use the crowd to answer DB-hard queries

Where to use the crowd:

- **Find missing data**
- **Make subjective comparisons**
- **Recognize patterns**

But not:

- Anything the computer already does well



M. Franklin, D. Kossmann, T. Kraska, S. Madden, S. Ramesh, R. Xin. CrowdDB: Answering Queries with Crowdsourcing, *SIGMOD 2011*

CrowdSQL

DDL Extensions:

Crowdsourced columns

```
CREATE TABLE company (
  name STRING PRIMARY KEY,
  hq_address CROWD STRING);
```

Crowdsourced tables

```
CREATE CROWD TABLE department (
  university STRING,
  department STRING,
  phone_no STRING)
PRIMARY KEY (university, department);
```

DML Extensions:

CrowdEqual:

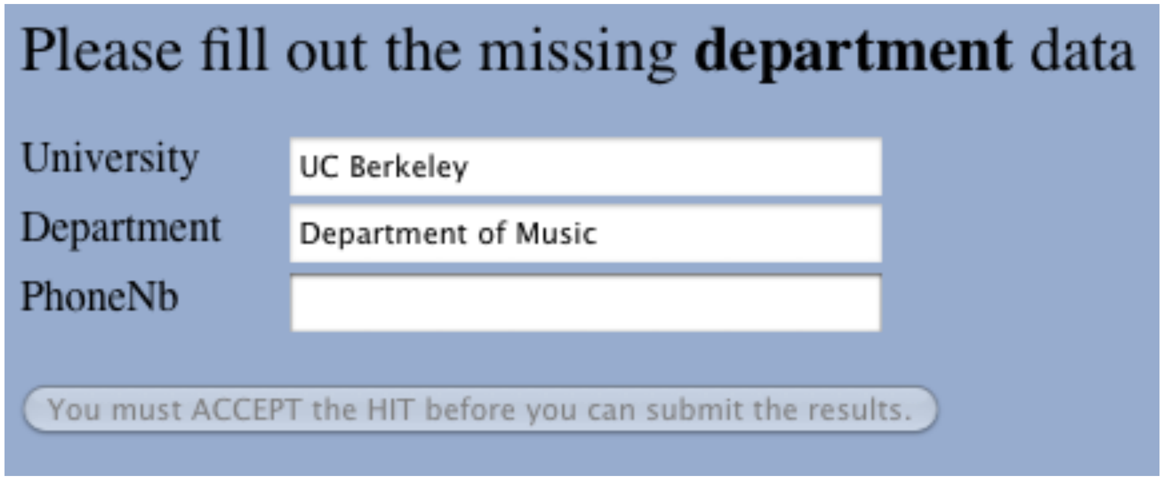
```
SELECT *
FROM companies
WHERE Name ~= "Big Blue"
```

CROWDORDER operators (currently UDFs):

```
SELECT p FROM picture
WHERE subject =
    "Golden Gate Bridge"
ORDER BY CROWDORDER(p, "Which
pic shows better %subject");
```

User Interface Generation

- A clear UI is key to response time and answer quality.
- Can leverage the SQL Schema to auto-generate UI (e.g., Oracle Forms, etc.)



Please fill out the missing **department** data

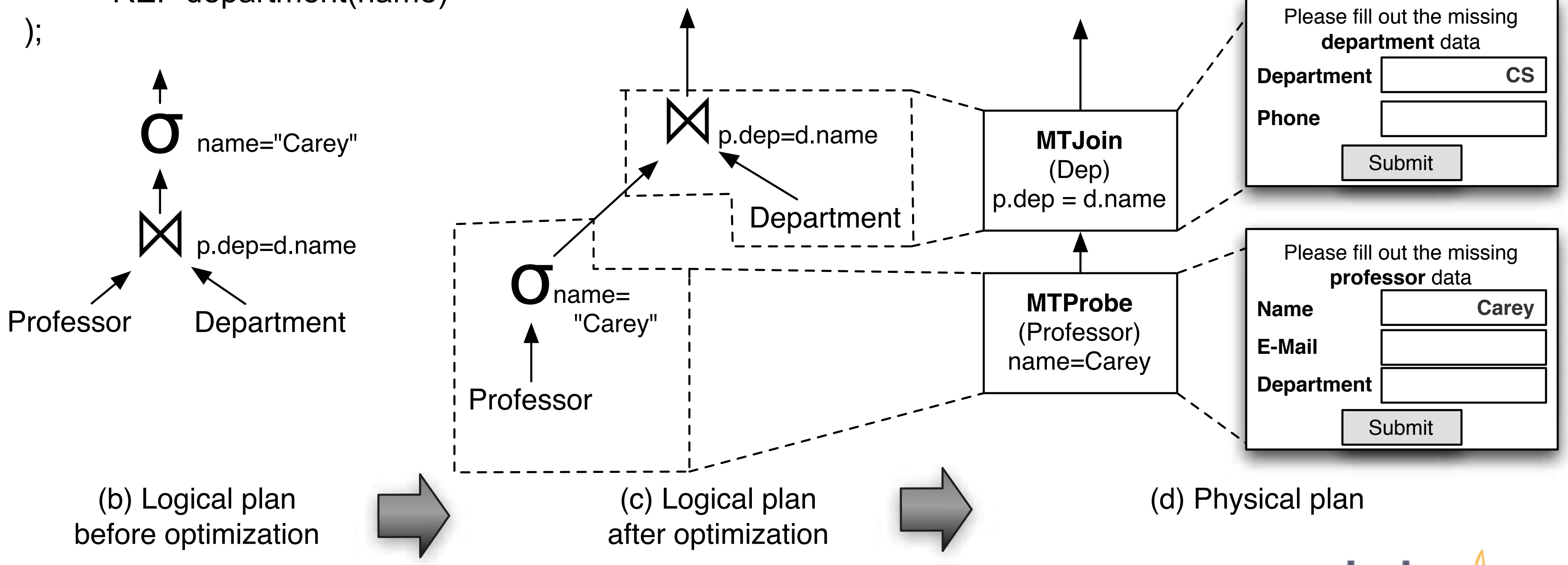
University	UC Berkeley
Department	Department of Music
PhoneNb	

Query Optimization and Execution

```
CREATE CROWD TABLE department (
   name STRING PRIMARY KEY
   phone_no STRING);

CREATE CROWD TABLE professor (
   name STRING PRIMARY KEY
   e-mail STRING
   dep STRING
     REF department(name)
);
```

```
SELECT *
FROM PROFESSOR p,
DEPARTMENT d
WHERE d.name = p.dep
AND p.name ="Michael J. Carey"
```



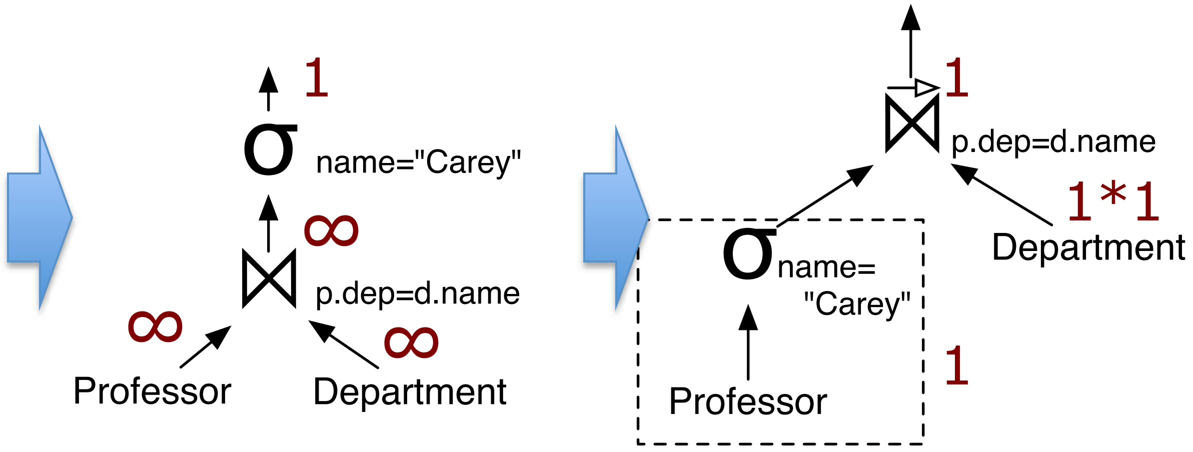
(b) Logical plan before optimization

(c) Logical plan after optimization

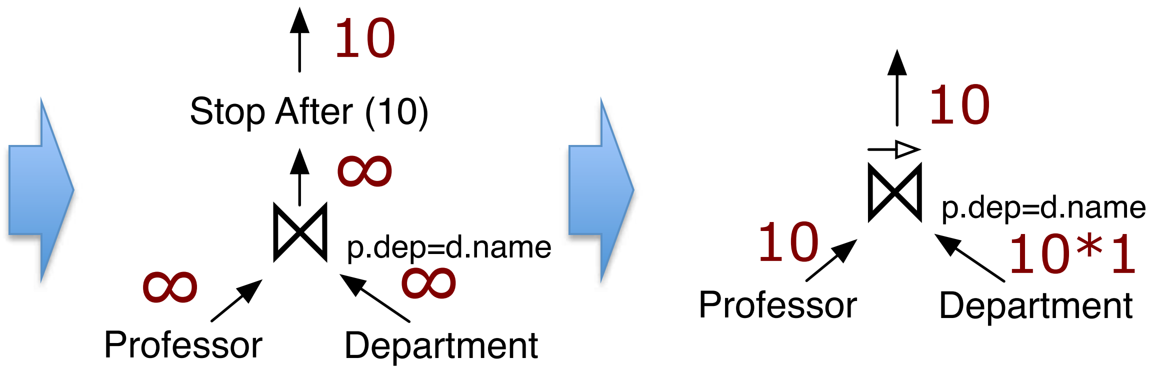
(d) Physical plan

Dealing with the Open-World

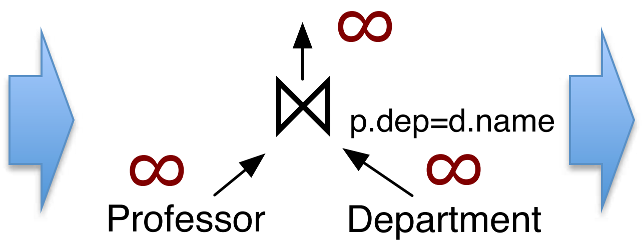
```
SELECT *
FROM PROFESSOR p,
  DEPARTMENT d
WHERE p.dep = d.name
AND p.name ="Carey"
```



```
SELECT *
FROM PROFESSOR p,
  DEPARTMENT d
WHERE p.dep = d.name
LIMIT 0, 10
```



```
SELECT *
FROM PROFESSOR p,
  DEPARTMENT d
WHERE p.dep = d.name
```



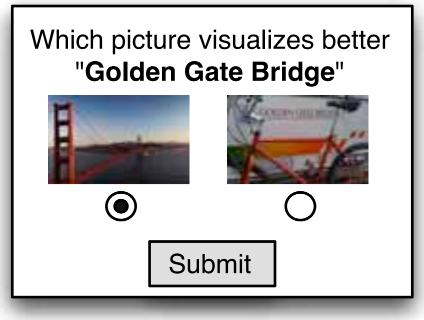
! Only allowed with iterative query improvement

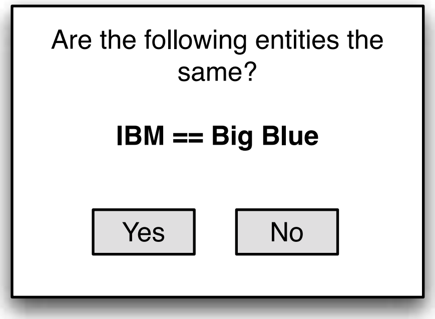
amplab

Subjective Comparisons

MTFunction

- implements the CROWDEQUAL and CROWDORDER comparison
- Takes some description and a type (equal, order) parameter
- Quality control again based on majority vote
- Ordering can be further optimized (e.g., Three-way comparisons vs. Two-way comparisons)


Which picture visualizes better "**Golden Gate Bridge**"

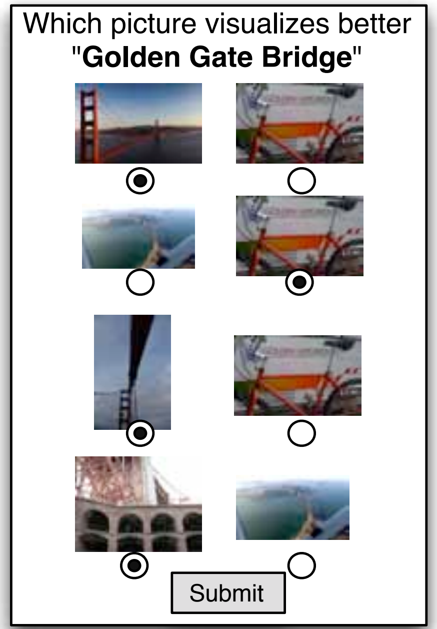

Are the following entities the same?

IBM == Big Blue

Does it Work?: Picture ordering

Query:

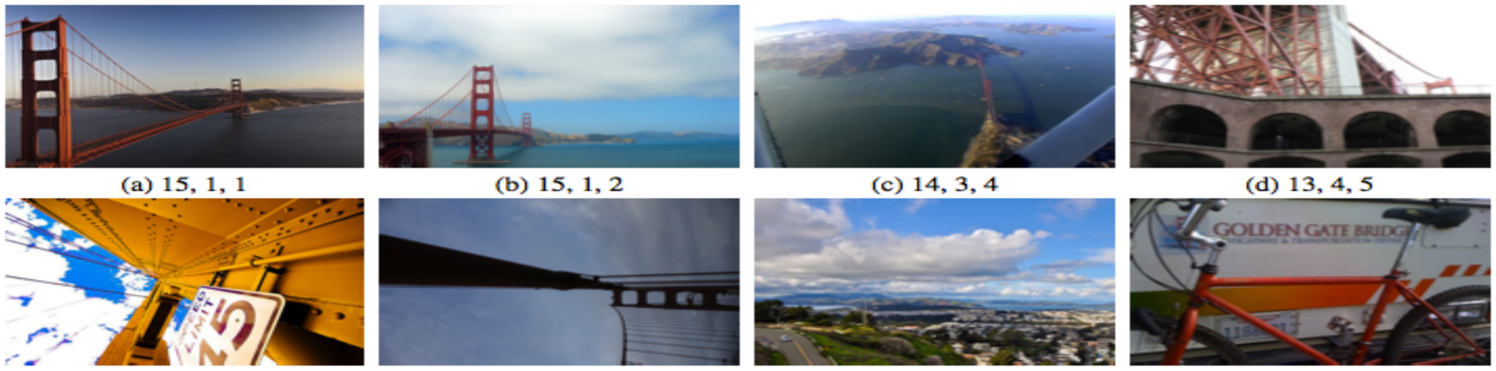
```
SELECT p FROM picture
WHERE subject = "Golden Gate Bridge"
ORDER BY CROWDORDER(p, "Which pic shows
                           better %subject");
```



Which picture visualizes better "**Golden Gate Bridge**"

Submit

Data-Size: 30 subject areas, with 8 pictures each
Batching: 4 orderings per HIT
Replication: 3 Assignments per HIT
Price: 1 cent per HIT



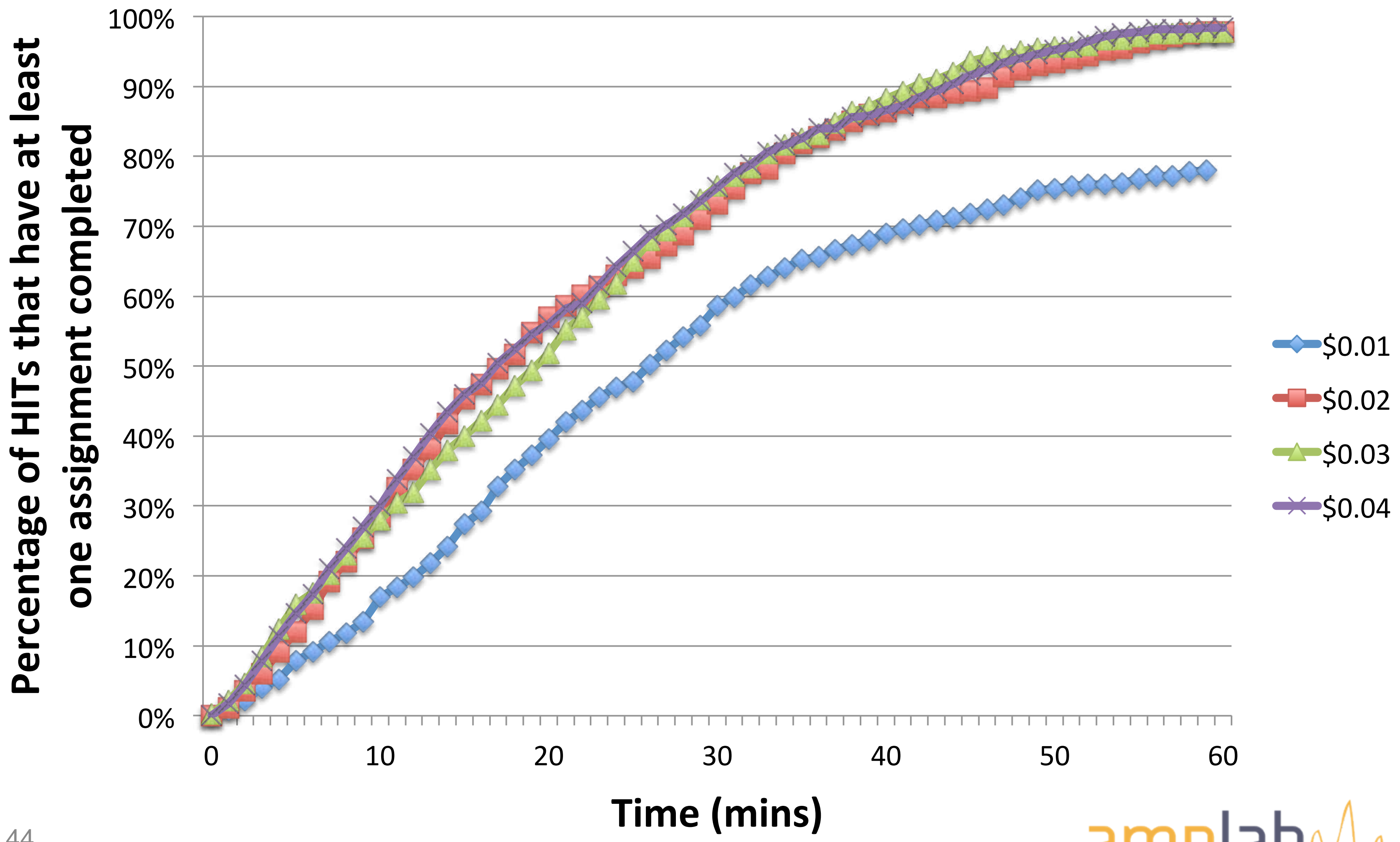
(a) 15, 1, 1 (b) 15, 1, 2 (c) 14, 3, 4 (d) 13, 4, 5
(e) 10, 5, 6 (f) 9, 6, 3 (g) 4, 7, 7 (h) 4, 7, 8

(turker-votes, turker-ranking, expert-ranking)

Can we build a "Crowd Optimizer"?

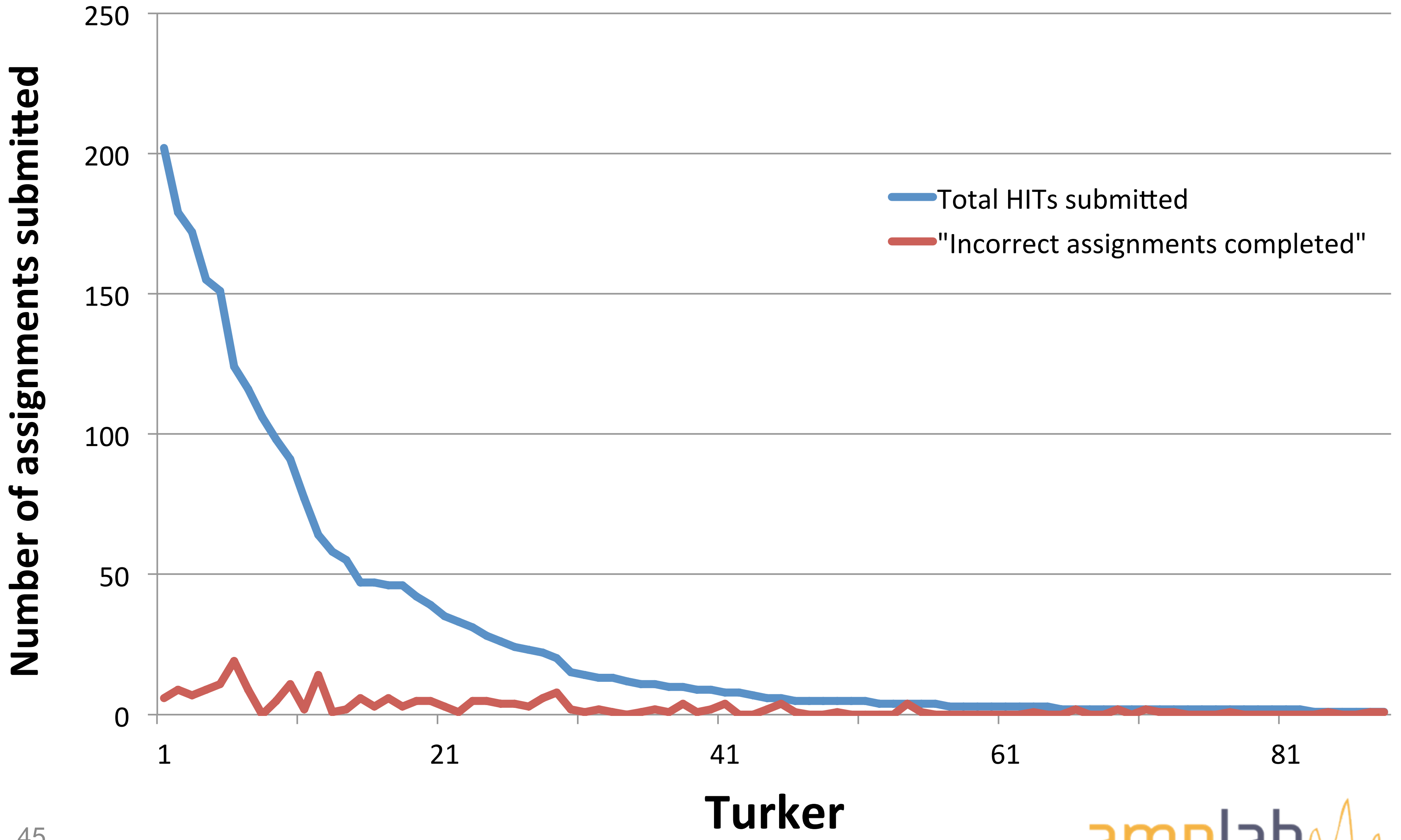
Price vs. Response Time

5 Assignments, 100 HITs

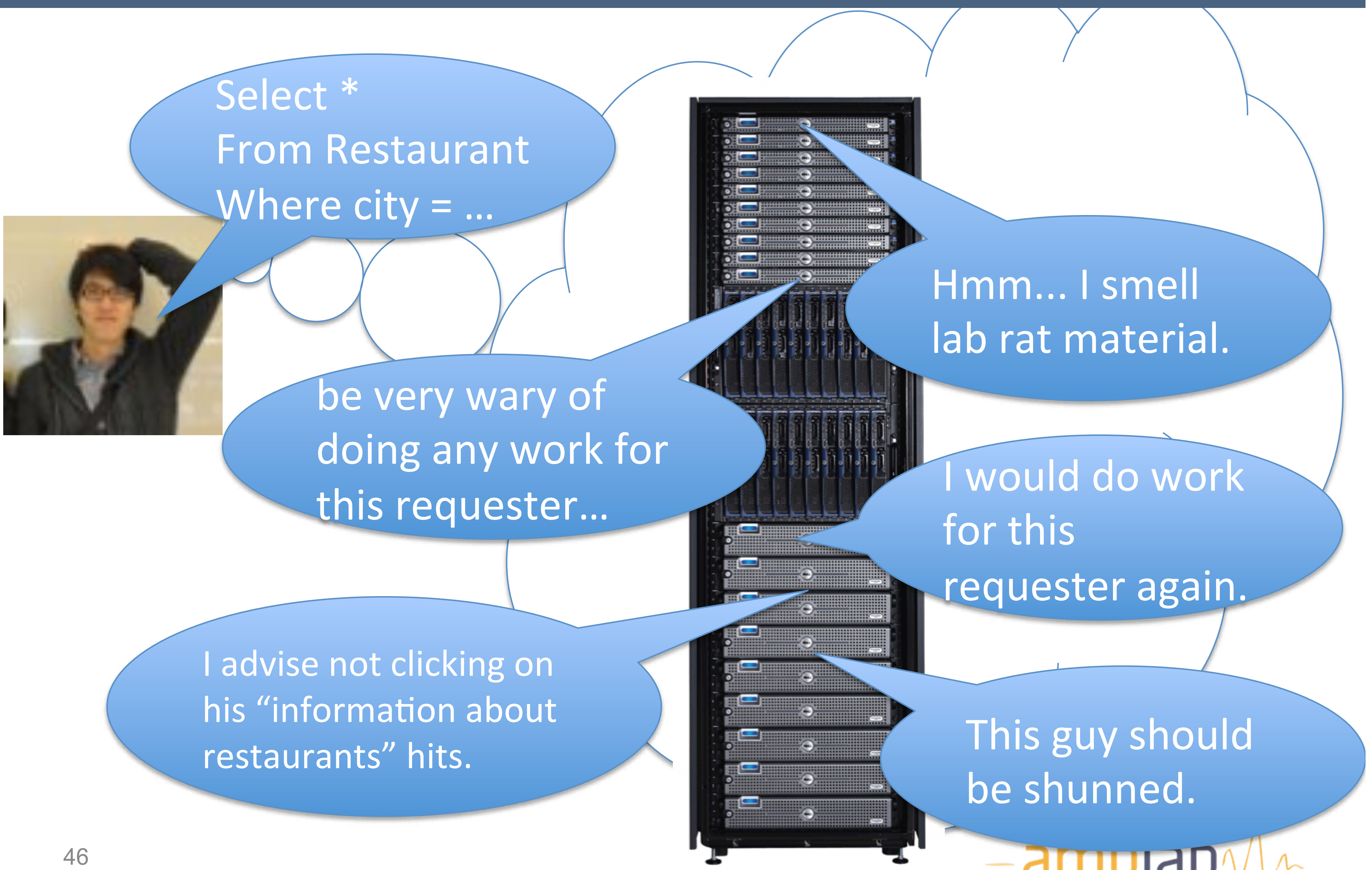


Turker Affinity and Errors

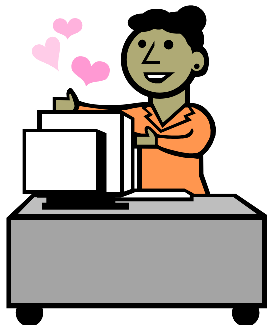
5 Assignments



Can we build a “Crowd Optimizer”?


Select *
From Restaurant
Where city = …
Hmm... I smell lab rat material.
be very wary of doing any work for this requester…
I would do work for this requester again.
I advise not clicking on his “information about restaurants” hits.
This guy should be shunned.

Processor Relations?



AMPLab

HIT Group » Simple straight-forward HITs, find the address and phone number for a given business in a given city. All HITs completed were approved. Pay was decent for amount of time required, when compared to other available HITs. But not when looked at from an hourly wage perspective. I would do work for this requester again. posted by…

fair:5 / 5 fast:5 / 5 pay:4 / 5 comm:0 / 5

Tim Klas Kraska

HIT Group » I recently did 299 HITs for this requester.… Of the 299 HITs I completed, 11 of them were rejected without any reason being given. Prior to this I only had 14 rejections, a .2% rejection rate. I currently have 8522 submitted HITs, with a .3% rejection rate after the rejections from this requester (25 total rejections). I have attempted to contact the requester and will update if I receive a response. Until then be very wary of doing any work for this requester, as it appears that they are rejecting about 1 in every 27 HITs being submitted. posted by …

Crowd as specialized CPUs

	Cloud	Crowd
Cost	($0.02 - $2.10)/ hour	$4.8 / hour
Pay Model	pay-as-you-go	pay-as-you-go
Investment	none	training
Response Time	Varied: msec	Varied: min, hours, days
Capabilities / Features	good (number crunching) & bad (AI)	good (AI) & bad(number crunching)
Programming	formal	natural language GUI
Availability	virtually unlimited	???
Affinity	virtualized	personal
Reliability	Faulty but not malicious	Faulty and possibly malicious
Legal	Privacy	Privacy++, taxes, benefits,..

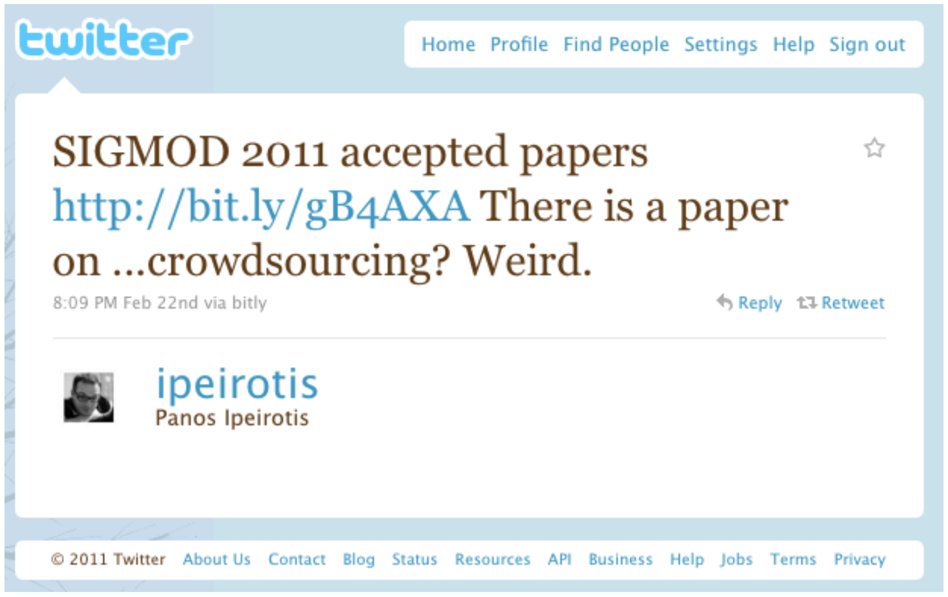
Future: Crowdsourcing → DB++?

- Cost Model for the Crowd
  - Latency (mins) vs. Cost ($) vs. Quality (%error)
- Adaptive Query Optimization
  - How to monitor crowd during query execution?
  - How to adapt the query plan?
- Caching / Materializing Crowd Results
  - E.g., maintaining the cached values
- Complexity Theory
  - Three-way comparisions vs. Two-way comparisons
- Privacy: Public vs. Private crowds
  - Flip-side of affinity of turkers
- Meta-crowds: Crowds help crowds (e.g. UI)

Future: DB → Crowdsourcing++?

Crowd-Hard Problems:

- Programming Language: GUI
- Many, many knobs to turn
- Changing platform behavior
- Quality Control
- Learning effects
- Community Management
- …


twitter
Home Profile Find People Settings Help Sign out

SIGMOD 2011 accepted papers http://bit.ly/gB4AXA There is a paper on ...crowdsourcing? Weird.

8:09 PM Feb 22nd via bitly

ipeirotis
Panos Ipeirotis

© 2011 Twitter About Us Contact Blog Status Resources API Business Help Jobs Terms Privacy

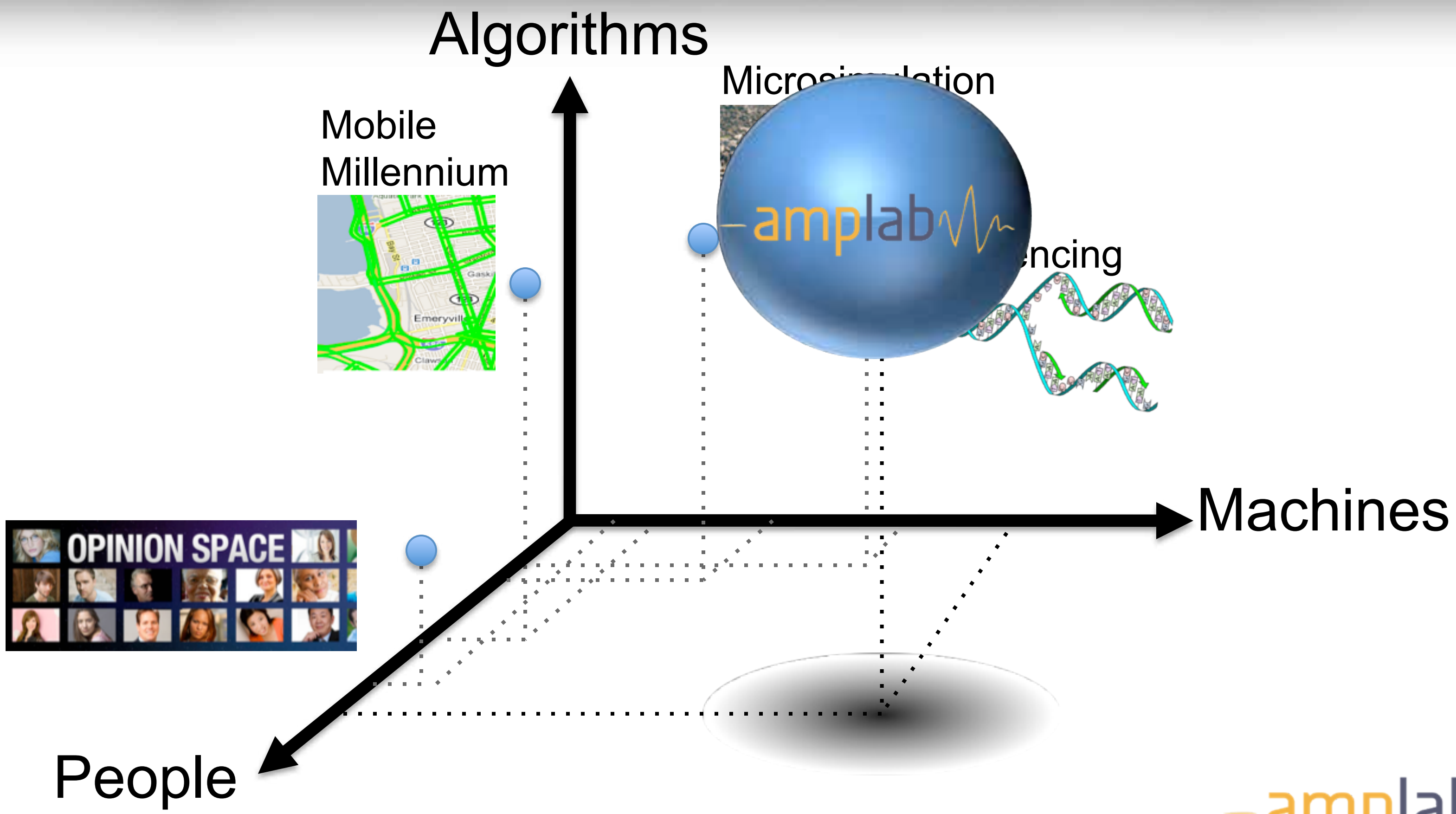
The DB-Approach can help?

- Data independence, cost-based optimization, schema management…

amplab

The AMPLab

Make sense at scale by holistically integrating Algorithms, Machines, and People



amplab

Summary - AMPLab

- Goal: Tame Big Data Problem

  Balance **quality**, **cost** and **time** to solve a given problem
- *The* computing challenge of the decade
- Widespread applicability across applications
- To address, we must Holistically integrate

  Algorithms, **M**achines, and **P**eople

We thank our sponsors:

Google SAP amazon.com web services cloudera ERICSSON HUAWEI IBM intel Microsoft NetApp NEC ORACLE splunk> vmware

amplab