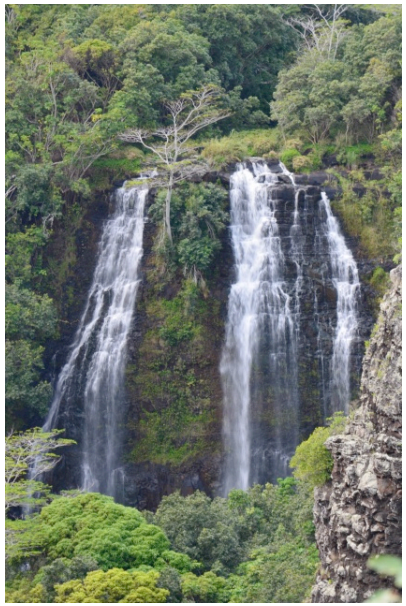
Big Data Research: Will Industry Solve all the Problems?

Magdalena Balazinska

Department of Computer Science & Engineering
University of Washington

The Exciting Times of "Big Data"

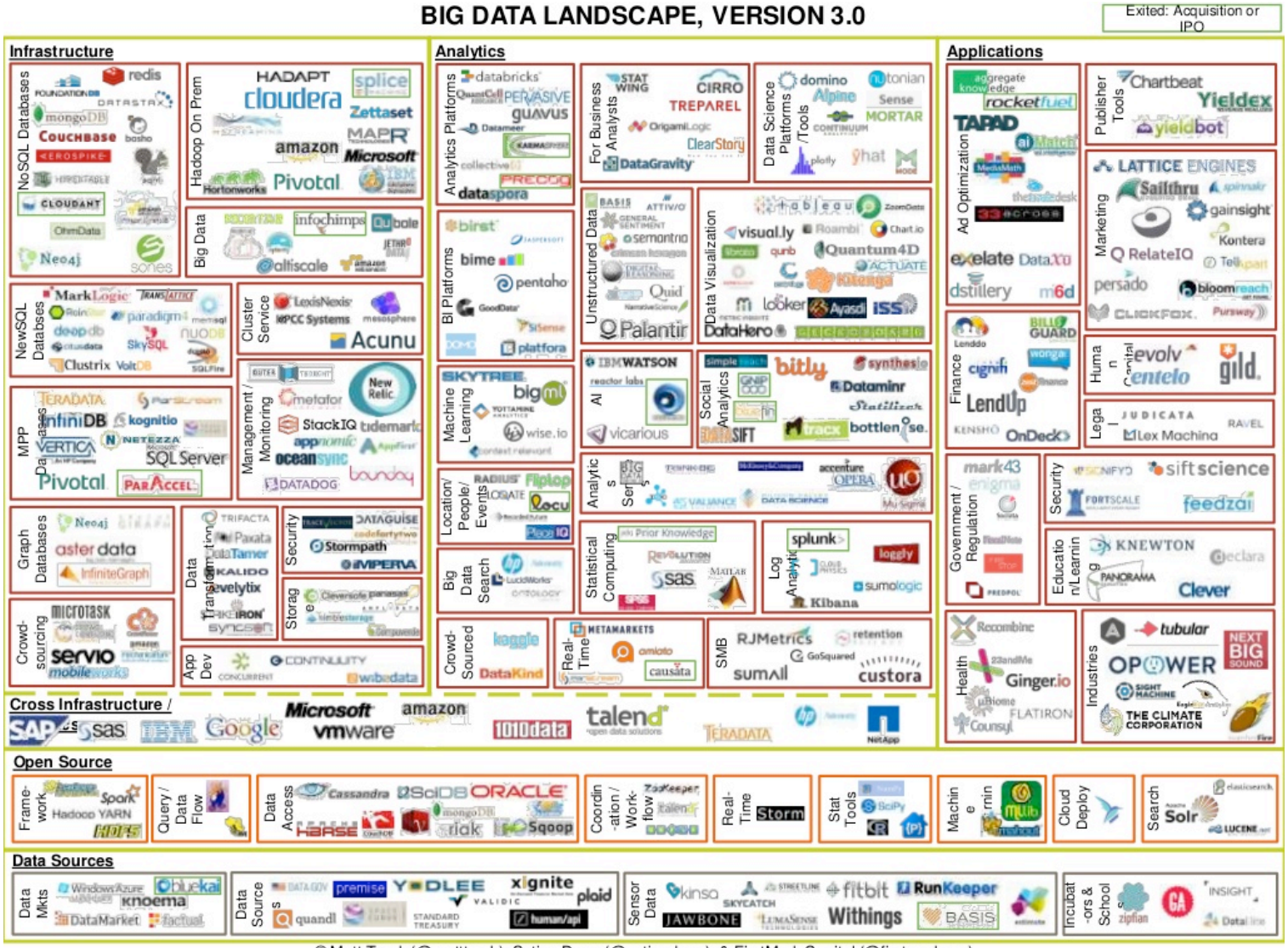


Everyone today has a big data problem

- Whether it is a data lake, data swamp, or data stream
- Whether they call it big data, data science, data wrangling, ..

Photo by Gary Bridgman / CC BY

Tremendous Activity in Industry



© Matt Turck (@mattturck), Sutian Dong (@sutiandong) & FirstMark Capital (@firstmarkcap)

How can academia contribute?

VLDB Has a Long History in Big-Data Research

VLDB 1975

A COMPUTER ARCHITECTURE FOR LARGE (DISTRIBUTED) DATA BASES

Richard Peebles and Eric Manning

Dept. of Computer Science
Computer Communications Networks Group
University of Waterloo

ABSTRACT

It is argued that the data-base of a nation-wide organization will exhibit geographic locality of reference. That is, most of the transactions homing on a given component of the data base originate from a particular geographic region. At the same time there is a need to operate the collection of components as a single data base to provide for occasional transactions which cross regional boundaries, and

Distributed Databases an Early Theme

Concurrency Control in Distributed Systems

VLDB 1980

- Rudolf Bayer, Klaus Elhardt, Hans Heller, Angelika Reiser:
  Distributed Concurrency Control in Database Systems. 275-284
- Philip A. Bernstein, Nathan Goodman:
  Timestamp-Based Algorithms for Concurrency Control in Distributed Database Systems. 285-300
- Wing Kai Cheng, Geneva G. Belford:
  Update Synchronization in Distributed Databases. 301-308

Data Base Transactions

VLDB 1981

- Jim Gray:
  The Transaction Concept: Virtues and Limitations (Invited Paper). 144-154
- Andrea J. Borr:
  Transaction Monitoring in ENCOMPASS: Reliable Distributed Transaction Processing. 155-165
- Catriel Beeri, Ron Obermarck:
  A Resource Class Independent Deadlock Detection Algorithm. 166-178

Parallel Processing Also Early

VLDB 1982

Parallel Algorithms and Their Implementation in MICRONET*

Stanley Y. W. Su
Krishna P. Mikkilineni

Database Systems Research and Development Center
Department of Computer and Information Sciences
University of Florida

Abstract

This paper describes a simple microcomputer network system and its architectural support for four categories of database operations. The design and implementation of hardware and software and the parallel algorithms for the database operations are described and illustrated. Three tion of hardware and algorithm designs are of paramount importance to achieve the needed efficiency for handling database problems.

This paper deals with the use of a simple and flexible microcomputer network (MICRONET) for the implementation of four categories of algorithms useful for database management. It describes the architectural supports for the

Parallel Processing Also Early

VLDB 1985

Multiprocessor Hash-Based Join Algorithms

David J. DeWitt
Robert Gerber

Computer Sciences Department
University of Wisconsin

ABSTRACT

This paper extends earlier research on hash-join algorithms to a multiprocessor architecture. Implemen-

papers reached the same conclusion: that while sort-merge is the commonly accepted algorithm for ad-hoc joins, it is, in fact, not nearly as fast as several join algorithms based on hashing. In retrospect, it is interesting to observe that a simple, but very good algo-

VLDB 1986

GAMMA - A High Performance Dataflow Database Machine

David J. DeWitt
Goetz Graefe
Krishna B. Kumar
Robert H. Gerber
Michael L. Heytens
M. Muralikrishna

Computer Sciences Department
University of Wisconsin

Other Big Data Topics in Early VLDB Days

- Sampling
  - Frank Olken, Doron Rotem: "**Simple Random Sampling from Relational Databases."** VLDB 1986.
- Active databases (ancestors of data streams)
  - A. Rosenthal, S. Chakravarthy, B. T. Blaustein, J. A. Blakeley: "**Situation Monitoring for Active Databases**." VLDB 1989
- Parallel, shared-nothing, in-memory processing
  - Annita N. Wilschut, Jan Flokstra, Peter M. G. Apers: **Parallelism in a Main-Memory DBMS: The Performance of PRISMA/DB.** VLDB 1992.
- Semistructured data (1998)

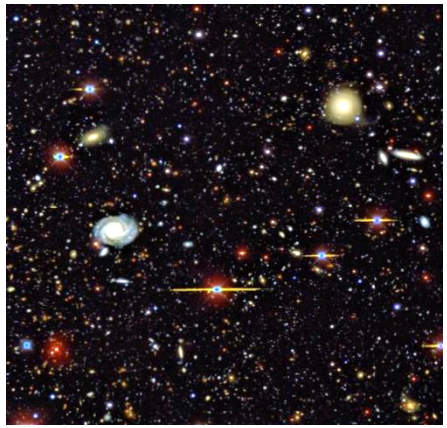
So what is next?

- Did we already solve all the big data problems?
  - Clearly not given all the innovation today
- Should we let industry take over?
  - Did we lay the foundation and our work is done?
  - Clearly not… we can contribute in fundamental ways

How can we contribute if industry has all the workloads?

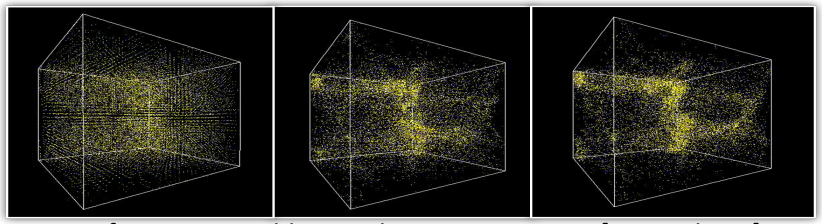
We Do Have Interesting Workloads

Images (telescope, satellite, …)



Picture from Deep Lens Survey (DLS: Tyson)

Spatio-temporal data (universe, oceans, …)

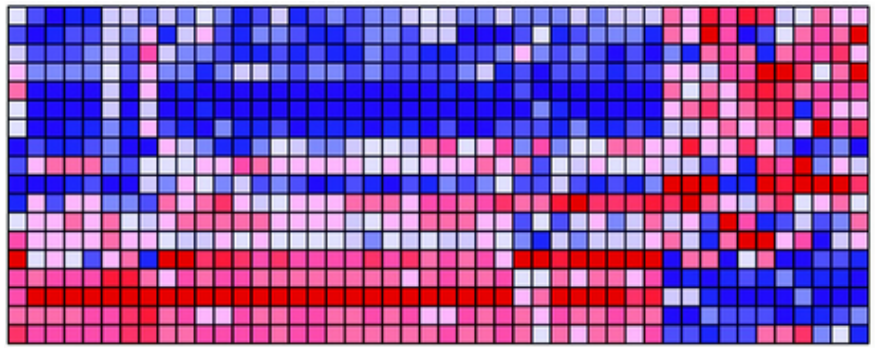


Picture from D. H. Stalder et. al. arXiv:1208.3444 [astro-ph.CO]

Large matrices (genome data, …)

>1M features from molecular snapshot

patients



Picture from Su-In Lee

Hadoop's Adolescence *[VLDB13]*

An analysis of Hadoop usage in scientific workloads

Kai Ren[1], YongChul Kwon[2], Magdalena Balazinska[3], Bill Howe[3]
[1] Carnegie Mellon University, [2] Microsoft, [3] University of Washington
kair@cs.cmu.edu, ykwon@microsoft.com, {magda,billhowe}@cs.washington.edu

ABSTRACT

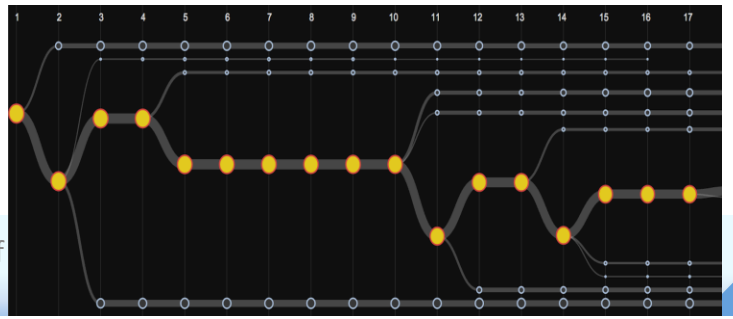
We analyze Hadoop workloads from three different research clusters from a user-centric perspective. The goal is to better understand data scientists' use of the system and how well the use of the system matches its design. Our analysis suggests that Hadoop usage is still in its adolescence. We see underuse of Hadoop features, extensions, and tools. We see significant diversity in resource usage and application styles, including some interactive and iterative workloads, motivating new tools in the ecosystem. We also observe significant opportunities for optimizations of these workloads. We find that job customization and configuration are used in

well Hadoop works for data scientists in terms of what they need to do to write Hadoop applications, execute them, tune them, and use them to extract knowledge from their data.

Our analysis is based on Hadoop workloads collected over periods of five to 20 months in three different clusters. Our traces comprise a total of more than 100,000 Hadoop jobs. The clusters that we study come from academic institutions. Our data scientists are 113 domain experts from various disciplines as we describe in more detail in Section 2. The dataset of one cluster is made publicly available for further study (at this URL: `www.pdl.cmu.edu/HLA/`).

The goal of our work is to better understand user behavior in a Hadoop cluster: Are users submitting workloads consistent with what Hadoop has been designed to handle? Does the MapReduce abstraction work well for the types of

Graphs (citations, galaxy evolutions, …)



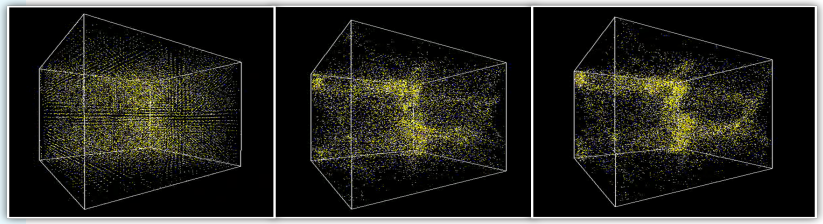
We Can Start With These Applications

Approach 1: Start with on-campus, scientific (or other) apps

- These apps have exciting and challenging requirements
- These requirements often generalize beyond campus

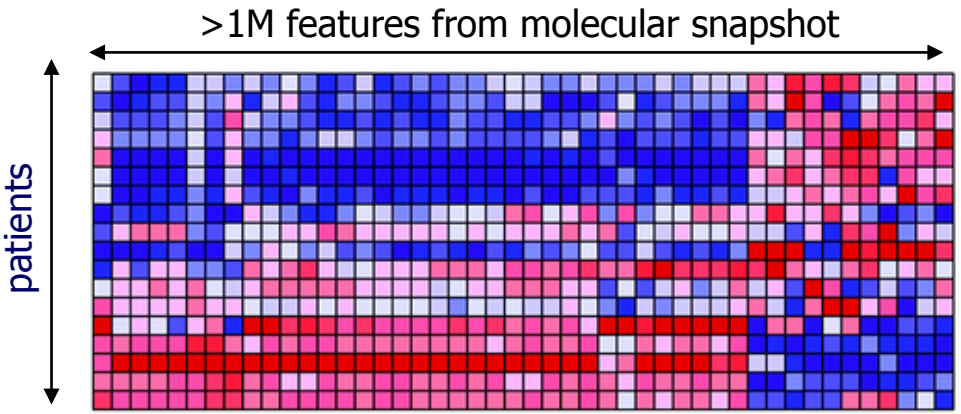
Telescope image:

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



N-body simulation data:

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



Genome data processing:

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

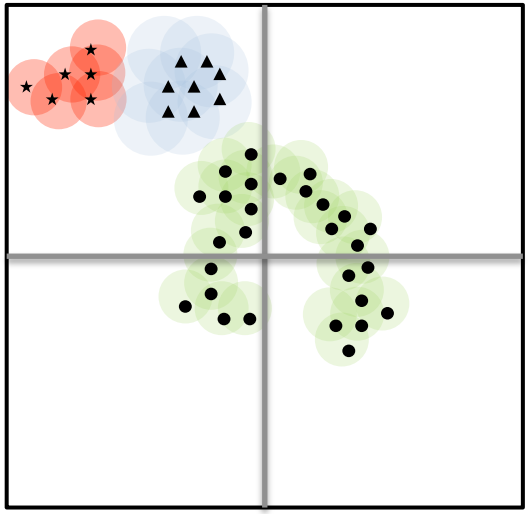
BUT industry has the latest big data systems!

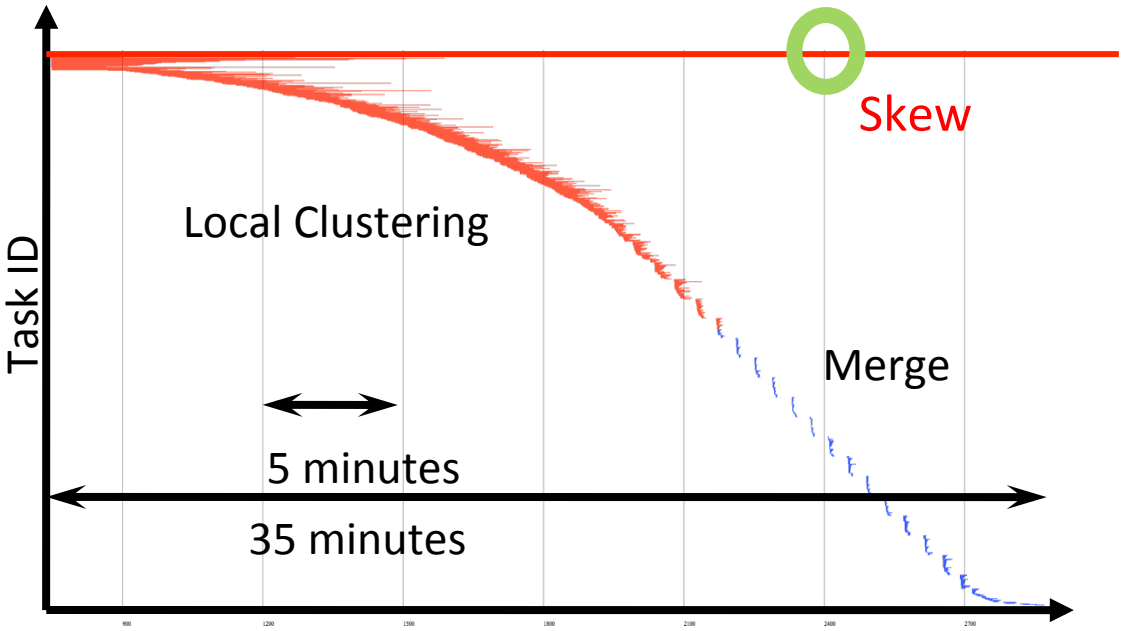
We Can Use Open-Source Tools

Approach 2: Apply open-source tools

- Most likely these tools will break in interesting ways
- Fix them and contribute back to open-source tools

Parallel data clustering
in Microsoft Dryad and Hadoop





Example: Getting HIGH PERFORMANCE is really hard!

We Can Use Open-Source Tools

Approach 2: Apply open-source tools

- Most likely these tools will break in interesting ways
- Fix them and contribute back to open-source tools

Developed **ParaTimer** [SIGMOD10]

- Shows progress of DAGs of Hadoop jobs

Developed **PerfXPlain** [VLDB12]

- Explains the performance of Hadoop jobs

Developed **SkewReduce** [SOCC10] and **SkewTune** [SIGMOD12]

- Based on Hadoop and available as open source

Developed **HaLoop** [VLDB10]

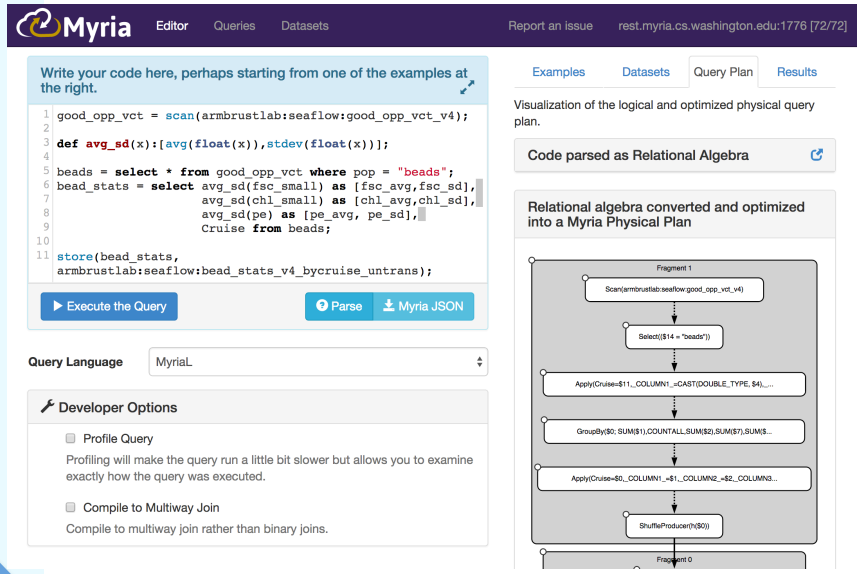
- Faster iterative processing in Hadoop also open source

We Can Invent New Tools

Approach 3: Build new tools

- Be creative
- Release tools as open source
- Offer them as a service

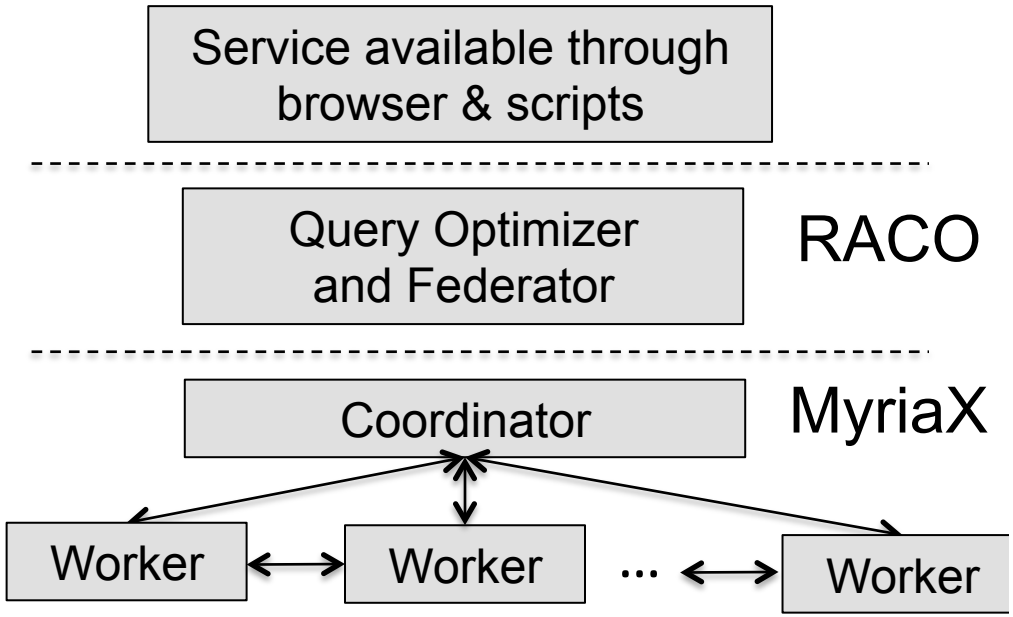
Example: Myria stack



Myria

http://myria.cs.washington.edu

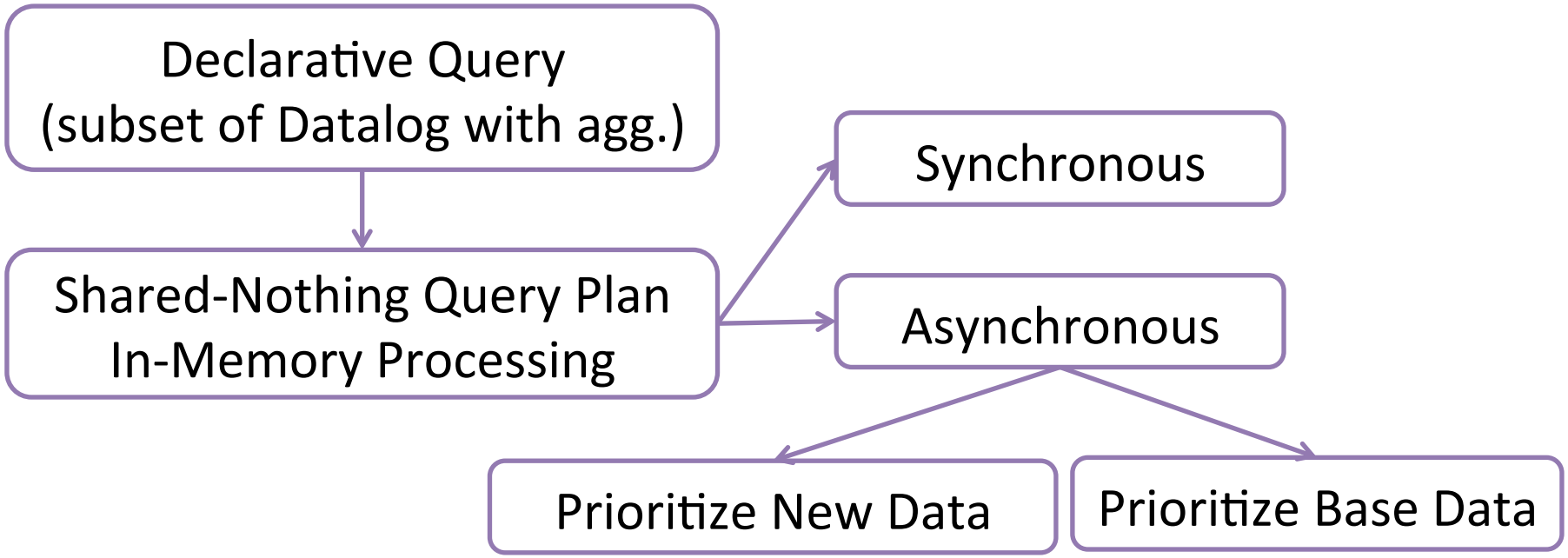
[SIGMOD14 Demo & VLDB15 Demo]



Internal distributed storage
Can also read from HDFS, S3, etc.
Queries can stretch to other engines

New Tools Lead to Rethinking Designs

Myria's advanced general-purpose iterative processing [VLDB15]



New Tools Lead to Rethinking Designs

Myria changes the face of cloud services [CIDR15]

Tier #1

Query Template	Runtime (seconds)
SELECT (9 ATTR.) FROM (PART) SELECT (9 ATTR.) FROM (CUSTOMER) SELECT (17 ATTR.) FROM (DATE) SELECT (60 ATTR.) FROM (5 TABLES) WHERE 0.1%	10
SELECT (17 ATTR.) FROM (LINEITEM) SELECT (9 ATTR.) FROM (2 TABLES) SELECT (3 ATTR.) FROM (5 TABLES) SELECT (60 ATTR.) FROM (5 TABLES) WHERE 10%	60
SELECT (60 ATTR.) FROM (5 TABLES)	300

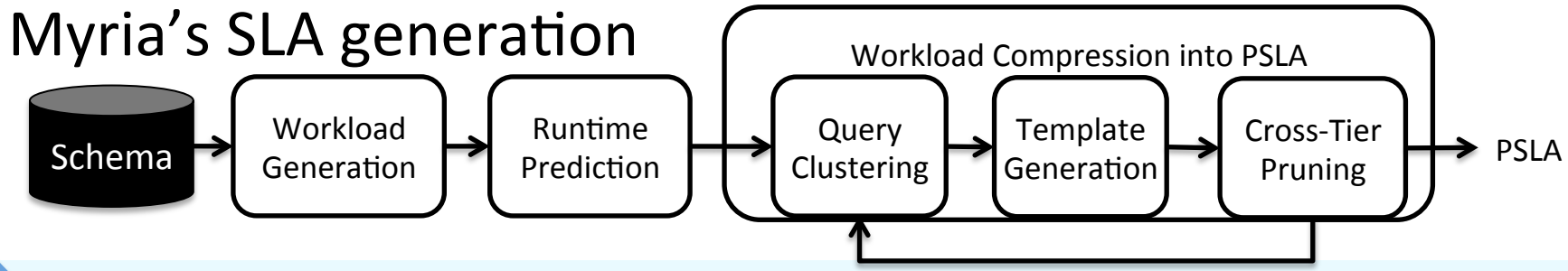
Purchase @ $0.16/hour

Tier #2

Query Template	Runtime (seconds)
SELECT (27 ATTR.) FROM (5 TABLES) WHERE 10% SELECT (60 ATTR.) FROM (5 TABLES) WHERE 1%	10
SELECT (11 ATTR.) FROM (2 TABLES) SELECT (9 ATTR.) FROM (5 TABLES)	60

Purchase @ $0.24/hour

Myria's SLA generation



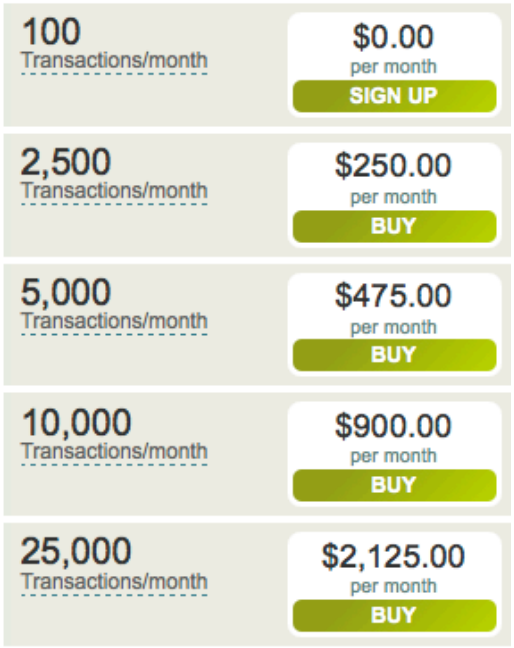
BUT I don’t want to be
in the heart of the race!

We Should Look Ahead

Approach 4: Consider problems on the horizon

- Data pricing (QueryMarket [SIGMOD13])
- Enforcing license agreements on data (DataLawyer [SIGMOD15])

How to price relational data?



Screenshot from the Azure Marketplace

How to automate data use agreements?

Terms of Service Subject to the terms and conditions ("Terms" or "Terms and Conditions") of this agreement ("Agreement"), you are granted a limited, nonexclusive license to use Versium services ("Versium Service" or "Service") and access the data ("Data"). For the purpose of this Agreement, Versium shall mean the Company and its parent corporate owner. The following Terms and Conditions govern the use of the Versium Service and the Data. By visiting Versium, accessing the Data or using the Service, you expressly agree to be bound by these Terms. 1. Limited License Permitted Use. You are granted personal, nontransferable and nonexclusive rights to access the Service and use the Data solely for your direct marketing, market research and customer prospecting purposes, in strict accordance with the Terms of the Agreement. Certain portions of the Data available through the Service are only available via license with use rights that are based upon subscription access. In such case of subs or termination of the subscription, and in such cas d by Versium, either (i) return the Data to Versium wi nformation thereon or (ii) provide a certificate, execute that the Data has been destroyed in such a manner t erable. (a) Your use of the Data will comply with all appli s and regulations ("Laws"), including Laws regarding tele citation. (b) Your use of any United States email Data SPAM Act, COPPA, and any State Registry Laws. (c) Versium reserves the right to review your use of the Data to ensure compliance with this Agreement, but any failure of Versium to review such use will not constitute acceptance of such use or waive any of Versium's rights hereunder or limit any of your obligations with respect to the Data. At any time upon at least three (3) days' notice, Versium may audit your records to determine whether you are in compliance with this Agreement and you will make available to Versium or its representatives all records necessary for the conduct of such an audit. Versium reserves the right to deny access to any user or group of users to the Versium Service, at its sole discretion, at any time, and for any reason or no reason. Versium reserves the right to remove any Data from the Versium database at any time and for any or no reason. Versium reserves the right to change, modify or otherwise alter these Terms and Conditions at any time at Versium's sole discretion. Any and all modifications shall become effective immediately once posted. You

Average length:
Over ***8*** pages!

BUT industry has more engineers!

We Educate the Next Generation

Approach 5: Prepare the next generation

- Prasang Upadhyaya -> Data Licensing -> Twitter
- Emad Soroush -> Array Database Engine -> GraphLab
- Nodira Khoussainova -> Big Data Usability -> Twitter
- YongChul Kwon -> Skew in Big Data Systems - > Microsoft
- Julie Letchner -> Probabilistic Streams -> Microsoft
- Evan Welbourne -> RFID Data Management -> Nokia Research

Conclusion

- We live in exciting times for data management
- Tremendous activity in industry
- Many ways for academia to contribute
  - Leverage workloads on academic campuses
  - Leverage open source software and contribute to it
  - Build new tools. Be creative!
  - Look ahead at problems looming on the horizon
  - Educate the next generation of researchers and engineers

- Acks: UW DB group, eScience and domain science collaborators, sponsors (NSF, Intel, Microsoft, Amazon, EMC, and Google)

Parallel Processing Also Early

VLDB 1979

concurrent

PARALLEL PROCESSING OF RELATIONS: A SINGLE-ASSIGNMENT APPROACH†

J. W. Schmidt
Fachbereich Informatik
Universität Hamburg
Schlüterstrasse 70
D-2000 Hamburg 13, West Germany

Abstract

Pascal/R, a language extension based on a data structure *relation* and some high level language constructs for relations [9] is augmented by a procedure concept for concurrent execution of database actions. Relation type procedure parameters serve two purposes: data accessing and access scheduling. Scheduling requirements are analyzed within the framework of the single-assignment approach [10] and proposals for the stepwise reduction of implementation effort are discussed.

general mechanism for selecting component relations used as parameters. The process of generating a single-assignment program out of individual action procedures is discussed in section 5. In section 6 restrictions imposed by a realistic implementation are analyzed.

In this paper we prefer the presentation of the overall concepts over a rigorous treatment of its details.

2. Relations and Subrelations