Data Management in the Cloud: Limitations and Opportunities

Daniel Abadi
Yale University
January 30th, 2009

Want milk with your breakfast?

- Buy a cow
  - Big upfront cost
  - Produces more (or less) milk than you need
  - Uses up resources
  - Time spent "maintaining it"
  - Unpleasant waste product
- Buy bottled milk
  - Continued cost
  - Buy what you need
  - Less resource intensive
  - No maintenance
  - Waste somebody else's problem

Your Computer is a Cow

- Your computer
  - Big upfront cost
  - Produces more (or less) "milk" than you need
  - Uses up resources (electricity)
  - Time spent maintaining it
  - Produces unpleasant waste (heat, noise)
- What if you could get computing power even more conveniently than bottled milk?

Cloud Computing is Bottled Milk

- Companies willing to rent computing resources from their data centers
- Resources include storage, processing cycles, software stacks
- Google, Microsoft, Amazon, Sun, Hewlett-Packard, Yahoo, EMC, and AT&T all taking part
- E.g., for $0.10/hour Amazon will give you:
  - 1.7 GB memory
  - Equivalent of 1.2 GHz processor
  - 350GB storage

Cloud Computing Concerns

- What if my data or service provider becomes unavailable?
- What if my supplier suddenly increases how much they charge me?
- What about security?
- What about lock in?

Cloud Computing Concerns

Remember: bottled milk is SOOO much cheaper and more convenient!

Key Cloud Characteristics for DBMS Deployment

- Compute power is elastic
  - But only if workload is parallelizable
  - Want shared-nothing DBMS
- Data is stored at an untrusted host
- Data is replicated, often across large geographic distances
  - Done under the covers
  - E.g., Amazon's "regions" and "availability zones"

Xactional DBMS Applications

- Problems:
  - Xactional DBMSs are typically not shared-nothing
  - It is hard to maintain ACID guarantees in the face of replication across large distances
    - CAP theorem: consistency, availability, tolerance to partitions … choose two
      - SimpleDB, PNUTS relax consistency
      - BigTable, Microsoft SQL Server Data Services relax atomicity
  - Large risks when storing operational data on an untrusted host

Analytical DBMS Applications

- Great fit for cloud deployment:
  - Shared-nothing is becoming standard
    - E.g., Teradata, Vertica, DATAllegro, Dataupia, Greenplum, Aster Data, DB2 DPF, Exadata, Netezza
  - ACID guarantees are not needed
  - Sensitive data can be left out of the analysis
- $5 billion market (1/3rd of DBMS market)

Cloud DBMS Wish List

- Efficiency
  - Pricing model makes this paramount
- Fault tolerance
  - Failures are common
  - Want no data loss
  - Want no work loss

Cloud DBMS Wish List

- Ability to run in a heterogeneous environment
  - It is nearly impossible to keep machines all running at the same speed
- Ability to interface with BI products
  - I.e. SQL, ODBC, JDBC interfaces
- Scale, scale, scale!

Data Analysis in the Cloud

- Parallel databases are the obvious choice right?
  - Interface with BI products
  - Compete fiercely on efficiency/performance
  - Scale horizontally

Parallel Database Scalability

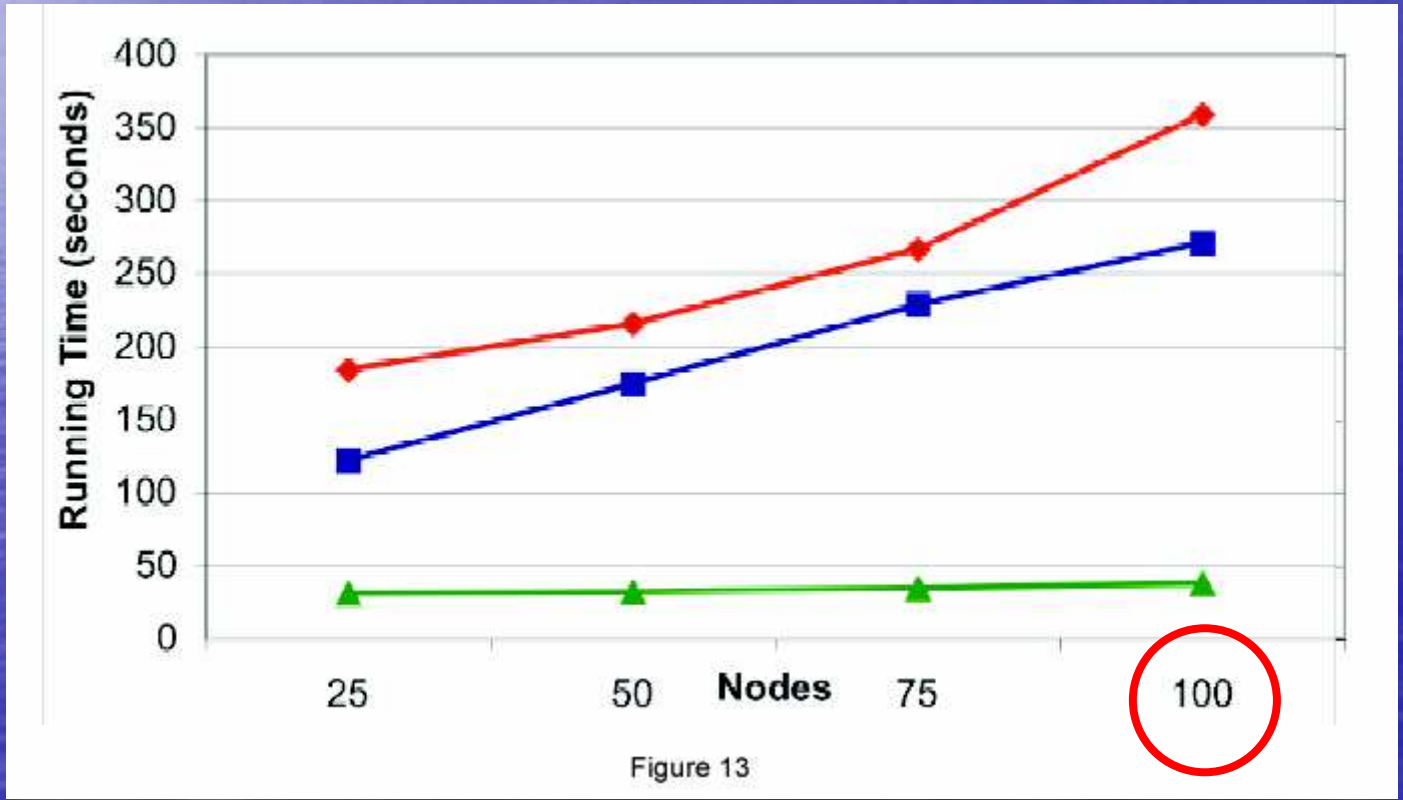


Figure 13

Parallel Database Scalability

- Try scaling them to 1000 nodes
  - Strange network bottlenecks
  - Restarting queries on a failure actually matter
  - Heterogeneous node effects

We want something …

- That can handle enormous scale …
- Does not restart queries upon a failure…
- Designed for heterogeneous environments…

- MapReduce?

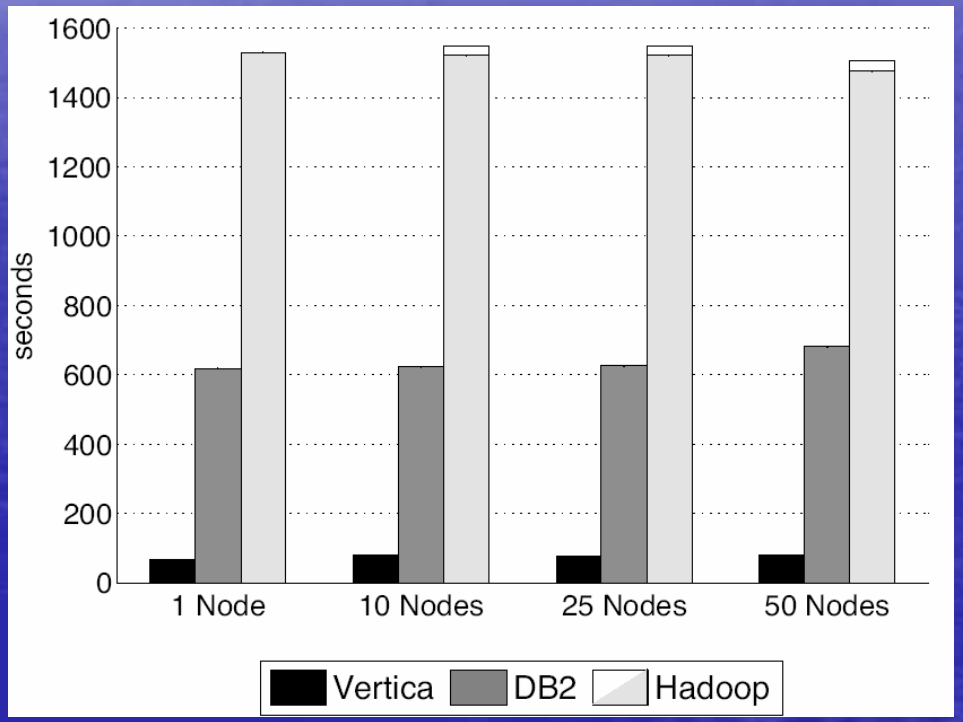
But MapReduce …

- Doesn’t interface with BI applications
- Is extremely inefficient

Efficiency

- CREATE TABLE UserVisits (
  sourceIP VARCHAR(16),
  destURL VARCHAR(100),
  visitDate DATE,
  adRevenue FLOAT,
  userAgent VARCHAR(64),
  countryCode VARCHAR(3),
  langCode VARCHAR(6),
  searchWord VARCHAR(32),
  duration INT);

- SELECT SUBSTR(sourceIP, 1, 7),
  SUM(adRevenue)
  FROM UserVisits
  GROUP BY SUBSTR(sourceIP, 1, 7);



Conclusion

- Data analysis well suited for the cloud
  - No current software meets all elements on wish-list
  - A hybrid between parallel databases and MapReduce is called for (Kamil Bajda-Pawlikowski and Azza Abouzeid to the rescue)

Come Join the Yale DB Group!