MAD Skills: New Analysis Practices for Big Data

Ronnie and Xuting

Some slides borrowed from a presentation on Joseph M. Hellerstein's page

What's new?

- Storage becomes cheap
- Growing number of massive-scale data: Clickstreams, software logs, email and forum archives
- Sophisticated data analysis leads to cost savings and even direct revenue!
- What's the solution? Greenplum!

Madgenda

- New requirements – MAD skills
- Fox Audience Network and challenges
- Overview of MAD model
- Supporting analytical functions in SQL
- A real MAD DBMS from Greenplum
- Questions we have

A hot job

"Looking for a career where your services will be in high demand?

… Provide a scarce, complementary service to something that is getting ubiquitous and cheap.

the sexy job in the next ten years will be statisticians

So what's ubiquitous and cheap? Data.

And what is complementary to data? Analysis.

Hal Varian, UC Berkeley, Chief Economist @ Google

MAD skills

- Magnetic
  attract data and analysts
- Agile
  rapid iteration
- Deep
  sophisticated analytics in Big Data
- In contrast with traditional Enterprise Data Warehouse and Business Intelligence.

Demanding Analysts

- Statisticians may have strong software skills.
- But would typically rather focus on deep data analysis.
- They wish to be freed from the work of DBA; need to be complemented by MAD data warehouse design.

Getting close to MAD (standard approaches)

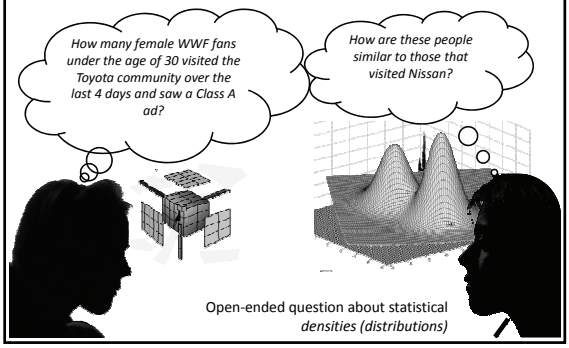
- Data Cubes, OLAP
  descriptive statistics; coarse overview
- Statistical packages (SAS, Matlab, R)
  takes huge time to load the data
  pushing the computation to data?
- Map/Reduce
  adopted by MAD
- Data mining algorithms
  targeted, black-box implementation
- Sloan Digital Sky Survey, SciDB

Fox Audience Network

- Greenplum DB
  - 42 Sun X4500s ("Thumper") *each* with:
    - 48 500GB drives
    - 16GB RAM
    - 2 dual-core Opterons
- Big and growing
  - 200 TB data (mirrored)
  - Fact table of 1.5 trillion rows
  - Growing 5TB per day
    - 4-7 Billion rows per day
- Variety of data
  - Ad logs, CRM, User data
- Research & Reporting
  - Diversity of users from Sales Acct Mgrs to Research Scientists
  - Microstrategy to command-line SQL
- Also extensive use of R and Hadoop

As reported by FAN, Feb, 2009

A Scenario from FAN



Open-ended question about statistical *densities (distributions)*

Challenges

- Huge data from different sources
- Disparate users pops up different questions (ranging from left to right)
- No set of predefined aggregates could cover every question
- R is very popular for multi-dimensional statistical analysis; however…

Getting real MAD

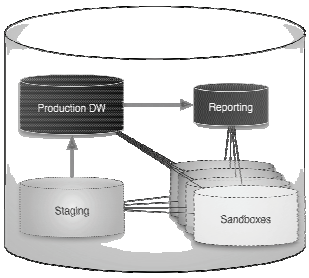
- Traditional Data Warehouse philosophy:
  There is no point in bringing data into the data warehouse environment without integrating it.
- It takes too much time!
- Analysts are tolerant to dirty data
- Satisfying the need of analysts is important and healthy

Virtuous Cycle of Analytics

- Analysts are not DBAs
  - They are data magnets
  - They tolerate and clean dirty data
  - They like *all* the data (no samples/extracts)
  - They *produce* data

Figure: *A Healthy Organization*

MAD Modeling



- Any comment on the model?

Statistical methods

- A hierarchy of mathematical concepts in SQL (MapReduce could also be used!)
- scalar -> vector -> function -> functional
- Encapsulated as stored procedures or UDFs
- Significantly enhance the vocabulary of the DBMS
- Requirements: focus on density methods; run massively parallel, on Big Data!

In the paper

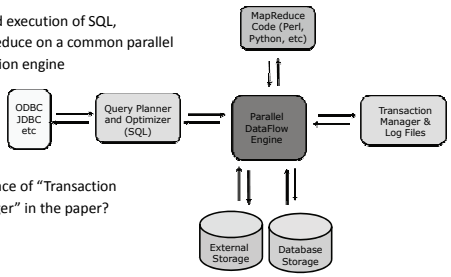
- Linear algebra (vectors/matrices)
- Ordinary Least Squares (multiple linear regression)
- Conjugate Gradiant (iterative optimization, e.g. for SVM classifiers)
- Functionals including Mann-Whitney U test, Log-likelihood ratios
- Resampling techniques, e.g. bootstrapping

Questions

- The paper presents operations written in SQL, any clue on Map/Reduce?
- There must be an optimizer, where?
- How is the final performance?
- It claims to run in parallel, how is it done? All we can see is normal SQL query.
- Not every query can be parallelized, so the programmer(analyst) need to think carefully.

Greenplum DB overview

- Unified execution of SQL, MapReduce on a common parallel execution engine



- Any trace of "Transaction Manager" in the paper?

MAD DBMS - Magnetic

- High speed loading – data-friendly
- Run queries on external tables
- Parallel accessing: Scatter/Gather Streaming requires external parallel feeding
- ETL vs. ELT (typo found!!)
- Parallel many-to-many loading architecture
- Promises: Automatic repartitioning of data from external sources; Performance scales with number of nodes; Negligible impact on concurrent database operations
- Example: 4 Tb/hour on FAN production system

MAD DBMS - Agile

- Multiple storage mechanism for different stages of data: data life cycle
- External storage/Greenplum AO storage
- Partitioning tables:
  partition exclusion;
  different storage format;
  atomic partition exchange

MAD DBMS - Deep

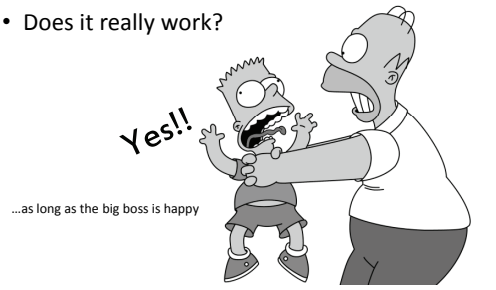
- Data analysts from different background:
  R, SAS, Matlab
  Java, Perl, Python
  How did they do this? Postgres.
- Map/Reduce programming interface
  AGAIN this is a simulation of Hadoop
- Attracts analysts; provides different programming styles; deep development

Future work

- Optimizing storage solution and queries at the same time (too many choices for now);
- Free analysts from physical design (data layout, writing queries without thinking parallel);
- Online query processing (similar to online MapReduce)

Interesting. But…

- Does it really work?



Interesting. But…

- Any technical details?
  eg. how are queries parallelized? Any optimization approaches? Execution engine?
- If we write in R (and other languages), it is not automatic parallel! Any clue about this?
- How does Map/Reduce provided here compare to Hadoop?
- Seems an ad for Greenplum?