Big Data Means at Least Three Different Things....

Michael Stonebraker
The Meaning of Big Data - 3 V’s

- **Big Volume**
  - With simple (SQL) analytics
  - With complex (non-SQL) analytics

- **Big Velocity**
  - Drink from a fire hose

- **Big Variety**
  - Large number of diverse data sources to integrate
Big Volume - Little Analytics

- Well addressed by data warehouse crowd
- Who are pretty good at SQL analytics on
  - Hundreds of nodes
  - Petabytes of data
In My Opinion....

- Column stores will win
- Factor of 50 or so faster than row stores
Big Data - Big Analytics

- Complex math operations (machine learning, clustering, trend detection, ...)  
  - the world of the “quants”  
  - Mostly specified as linear algebra on array data

- A dozen or so common ‘inner loops’  
  - Matrix multiply  
  - QR decomposition  
  - SVD decomposition  
  - Linear regression
Big Analytics on Array Data - An Accessible Example

- Consider the closing price on all trading days for the last 10 years for two stocks A and B.

- What is the covariance between the two time-series?

  \[
  (1/N) \times \sum (A_j - \text{mean}(A)) \times (B_j - \text{mean}(B))
  \]
Now Make It Interesting …

- Do this for all pairs of 4000 stocks
  - The data is the following 4000 x 2000 matrix

<table>
<thead>
<tr>
<th>Stock</th>
<th>$t_1$</th>
<th>$t_2$</th>
<th>$t_3$</th>
<th>$t_4$</th>
<th>$t_5$</th>
<th>$t_6$</th>
<th>$t_7$</th>
<th>...</th>
<th>$t_{2000}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_1$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$S_2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$S_{4000}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Hourly data? All securities?
Array Answer

• Ignoring the \((1/N)\) and subtracting off the means ....

\[
\text{Stock} \times \text{Stock}^T
\]
DBMS Requirements

- Complex analytics
  - Covariance is just the start
  - Defined on arrays
- Data management
  - Leave out outliers
  - Just on securities with a market cap over $10B
These Requirements Arise in Many Other Domains

- Auto insurance
  - Sensor in your car (driving behavior and location)
  - Reward safe driving (no jackrabbit stops, stay out of bad neighborhoods)
- Ad placement on the web
  - Cluster customer sessions
- Lots of science apps
  - Genomics, satellite imagery, astronomy, weather, ....
In My Opinion....

- The focus will shift quickly from “small math” to “big math” in many domains
- I.e. this stuff will become main stream....
Solution Options
R, SAS, MATLAB, et. al.

• Weak or non-existent data management

• File system storage

• R doesn’t scale and is not a parallel system
  – Revolution does a bit better
Solution Options
RDBMS alone

- SQL simulator (MadLib) is slooooow (analytics * .01)
  - And only does some of the required operations

- Coding operations as UDFs still requires you to simulate arrays on top of tables --- sloooow
  - And current UDF model not powerful enough to support iteration
Solution Options

R + RDBMS

- Have to extract and transform the data from RDBMS table to R data format
- ‘move the world’ nightmare
- Need to learn 2 systems
- And R still doesn’t scale and is not a parallel system
Solution Options
Hadoop

- Analytics * .01
- Data management * .01

- Because
  - No state
  - No “sticky” computation
  - No point-to-point messaging

- Only viable if you don’t care about performance
Solution Options

• New Array DBMS designed with this market in mind
An Example Array Engine DB
SciDB (SciDB.org)

- All-in-one:
  - data management on arrays
  - massively scalable advanced analytics

- Data is updated via time-travel; not overwritten
  - Supports reproducibility for research and compliance

- Supports uncertain data, provenance

- Open source

- Hardware agnostic
Big Velocity

- Trading volumes going through the roof on Wall Street - breaking infrastructure
- Sensor tagging of \{cars, people, \ldots\} creates a firehose to ingest
- The web empowers end users to submit transactions - sending volume through the roof
- PDAs lets them submit transactions from anywhere....
Two Different Solutions

- Big pattern - little state (electronic trading)
  - Find me a ‘strawberry’ followed within 100 msec by a ‘banana’

- Complex event processing (CEP) is focused on this problem
  - Patterns in a firehose

P.S. I started StreamBase but I have no current relationship with the company
Two Different Solutions

- Big state - little pattern
  - For every security, assemble my real-time global position
  - And alert me if my exposure is greater than X

- Looks like high performance OLTP
  - Want to update a database at very high speed
My Suspicion

- Your have 3-4 Big state - little pattern problems for every one Big pattern - little state problem
Solution Choices

- Old SQL
  - The elephants

- No SQL
  - 75 or so vendors giving up both SQL and ACID

- New SQL
  - Retain SQL and ACID but go fast with a new architecture
Why Not Use Old SQL?

• Sloooow
  – By a couple orders of magnitude

• Because of
  – Disk
  – Heavy-weight transactions
  – Multi-threading

• See “Through the OLTP Looking Glass”
  – VLDB 2007
No SQL

- Give up SQL
  - Interesting to note that Cassandra and Mongo are moving to (yup) SQL

- Give up ACID
  - If you need ACID, this is a decision to tear your hair out by doing it in user code
  - Can you guarantee you won’t need ACID tomorrow?
VoltDB: an example of New SQL

• A main memory SQL engine
• Open source
• Shared nothing, Linux, TCP/IP on jelly beans
• Light-weight transactions
  – Run-to-completion with no locking
• Single-threaded
  – Multi-core by splitting main memory
• About 100x RDBMS on TPC-C
In My Opinion

- ACID is good
- High level languages are good
- Standards (i.e. SQL) are good
Big Variety

• Typical enterprise has 5000 operational systems
  – Only a few get into the data warehouse
  – What about the rest?

• And what about all the rest of your data?
  – Spreadsheets
  – Access data bases
  – Web pages

• And public data from the web?
The World of Data Integration

the rest of your data

enterprise
data warehouse
text
Summary

- The rest of your data (public and private)
  - Is a treasure trove of incredibly valuable information
  - Largely untapped
Data Tamer

- Goal: integrate the rest of your data

- Has to
  - Be scalable to 1000s of sites
  - Deal with incomplete, conflicting, and incorrect data
  - Be incremental

- Task is never done
Data Tamer in a Nutshell

• Apply machine learning and statistics to perform automatic:
  – Discovery of structure
  – Entity resolution
  – Transformation

• With a human assist if necessary
  – WYSIWYG tool (Data Wrangler)
Data Tamer

- MIT research project
- Looking for more integration problems
  - Wanna partner?
Take away

• One size does not fit all

• Plan on (say) 6 DBMS architectures
  – Use the right tool for the job

• Elephants are not competitive
  – At anything
  – Have a bad ‘innovator’s dilemma’ problem
Newest Intel Science and Technology Center

- Focus is on “big data” - the stuff we have been talking about
  - Complex analytics on big data
  - Scalable visualization
  - Lowering the impedance mismatch between streaming and DBMSs
  - New storage architectures for big data
  - Moving DBMS functionality into silicon
- Hub is at M.I.T.
- Looking for more partners.....