



Ghosts of DBMSs Past: Lessons from your Parent's Big Data Wars

Paul G. Brown

Not for the First Time ...

An Explosion of Data Management Technology

Proclaiming obsolescence of everything prior

Promising simple/easy solutions to hard problems

Now in the “washing up” phase

NoSQL – “Not Yet” SQL, or “Don’t know SQL”

(Ex-)Hadoop – Everyone’s doing SQL

Necessity, and Sufficiency

What does a successful DBMS do (well) that’s necessary

Where SQL DBMSs might not be sufficient

Quality of Service

Schema

A file-system is an inadequate data model

Schema, and constraint rules, help data quality

Data store can exploit data semantics

Transactions (ACID)

Clear failure and concurrent access semantics

Multiple implementations; no-lock, ARIES, MVCC ...

Declarative Language Interfaces

No technical argument for SQL-like interfaces

End-user productivity. Value for \$. 'Nuff said ...

Is Set Theory Enough?

Encode, verses Measure

Traditional IT; humans encode information into data

Internet of Things: Machines measure phenomenon

“Electrical resistivity as temperature changes”

“Count of photons at various energy levels per second”

“Acoustic signal strength through some medium”

“Time; and therefore lat/long location”

All Subject to Observation Error

Measurements are random variables

SQL emphasizes equality/equivalent

Data management tools need statistical reasoning

Conclusion

Big Data Management Needs QoS

Garbage in, garbage out

File-systems provide a very low (inadequate) standard

Constraint rules, failure semantics, concurrent access

QoS is unrelated to data model, interfaces, etc

Is SQL Sufficient?

Language (and model) assume *encoded* information

IoT will produce random variables - *measurements*

Extend SQL with statistical reasoning?

Statistics makes heavy use of linear algebra (arrays?)