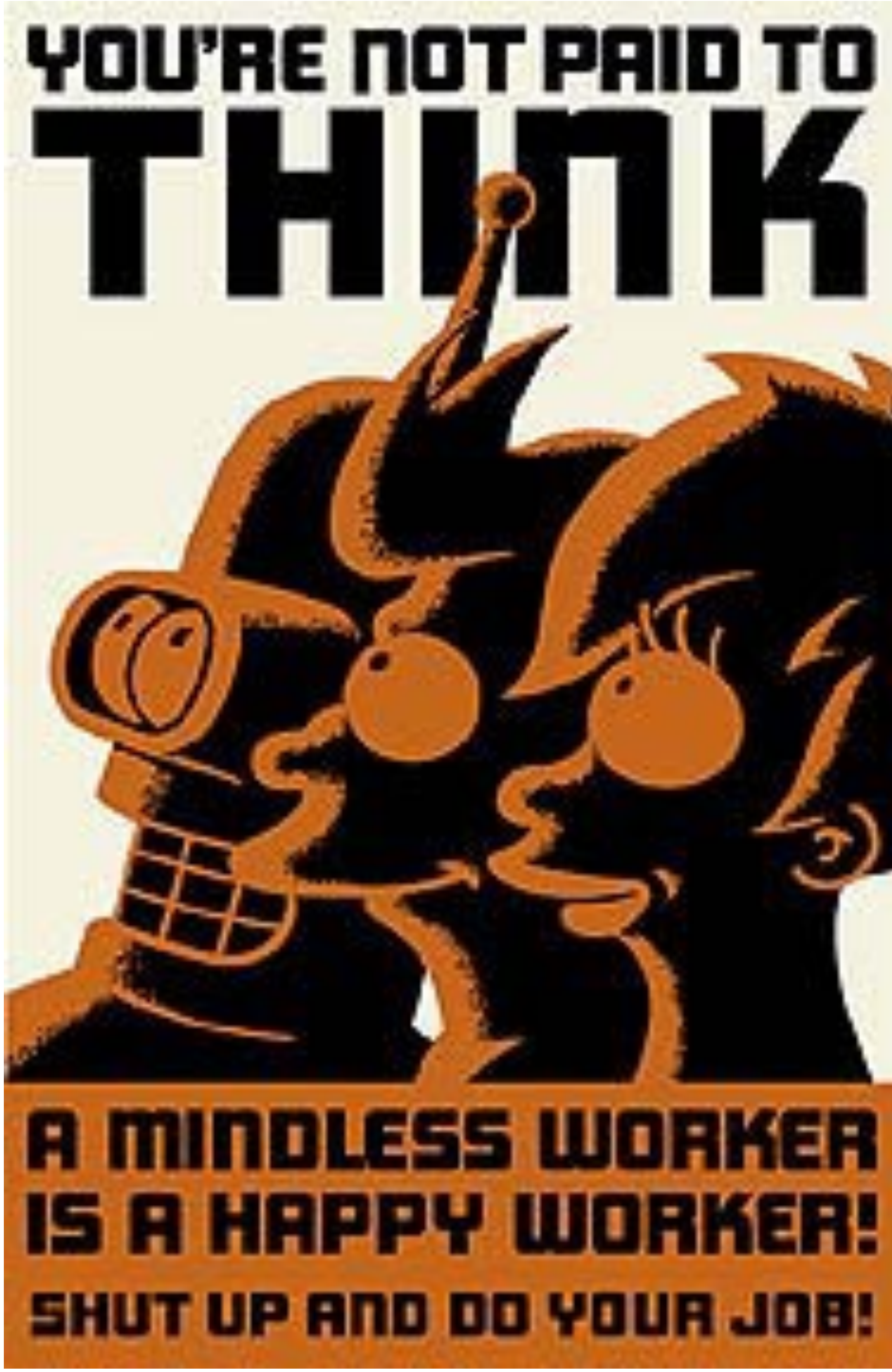


Following Google

Or

Don't Follow the Followers, Follow the Leaders

Or

The problem probably isn't the database, the problem is probably you

May, 2014

Mark Madsen
www.ThirdNature.net
@markmadsen

Third Nature

A Quick Intro

I may or may not be qualified to make the outlandish statements in this presentation. I have, however, made almost every mistake mentioned, and one learns by making mistakes.

You might say that's my singular skill.

History isn't taught in most university science curricula
(probably because it's a rabbit hole)

A BRIEF HISTORY OF DATA STORAGE AND RETRIEVAL

Third Nature

Databases: the problem statements over time

“Information has become a form of garbage, not only incapable of answering the most fundamental human questions but barely useful in providing coherent direction to the solution of even mundane problems.” – ***Neil Postman, 1985***

"We have reason to fear that the multitude of books which grows every day in a prodigious fashion will make the following centuries fall into a state as barbarous as that of the centuries that followed the fall of the Roman Empire.” – ***Adrien Baillet, 1685***

”…so many books that we do not even have time to read the titles." – ***Anton Francesco Doni, 1550***

Third Nature

The origin of information management problems

MS 4631
Bulla-envelope with 11 plain and complex tokens inside.
Near East, ca. 3700-3200 BC

For ~5000 years we used counters of various types, eventually developing writing to cope with civilization's needs.

Writing is more efficient than counters you can lose.

Sumerian bulla envelope with tokens.
The beta period.

Third Nature

Information Technology v1.0: Clay Tech, ~3000 bce

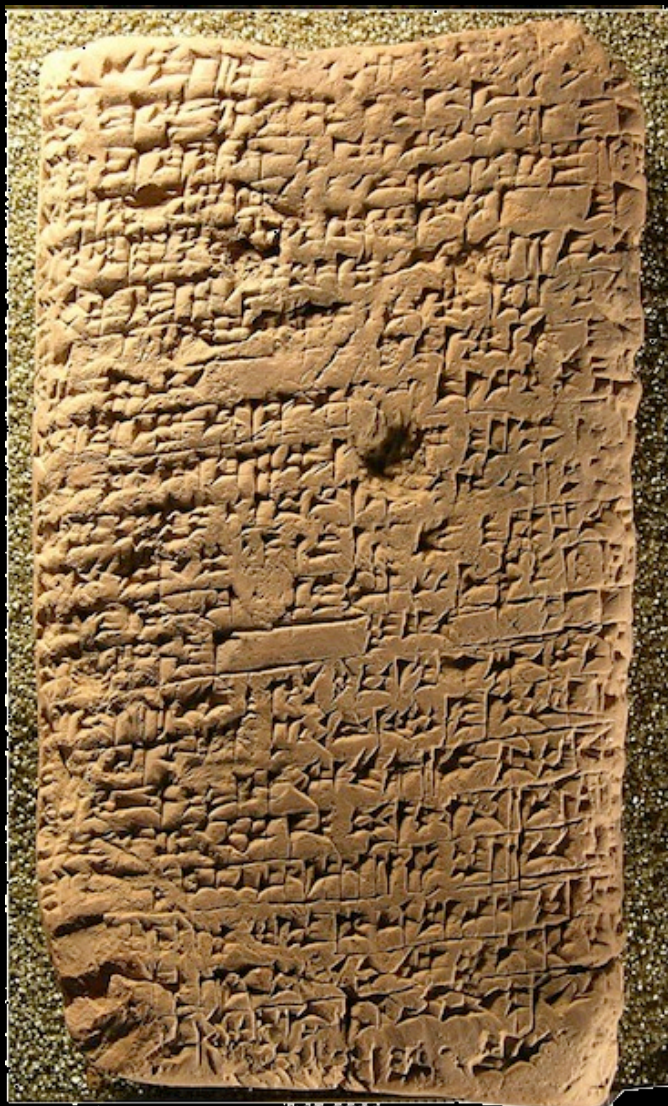
The first information explosion

That explosion led to the first metadata

Small piles in baskets are easy to tag and search

Third Nature

Metadata v1.0: tablets about tablets



MS 3391
Library catalogue. Babylonia, 2000-1600 BC

When there are enough of these lying around you need to work on organization of the collection by categorizations, aka “taxonomy”, “schema”

Like working out what tables are in a database, or what files are stored in HDFS.

Babylonian library catalog ~2000bce

Third Nature

Metadata v1.1: tablets about what's *in* tablets

When literacy rates are higher and people need to communicate more effectively, you need to invent mechanisms to cope, like dictionaries.

Now we're worried about what's inside the documents, not where they are placed.

Synonym list, Ashurbanipal, ~900 bce

Third Nature

Clay Tech has some familiar limitations

Information Management v2.0 Paper Tech*

Lighter, denser, faster storage media

Third Nature

More information = need for new metadata techniques: content tagging, author catalogs

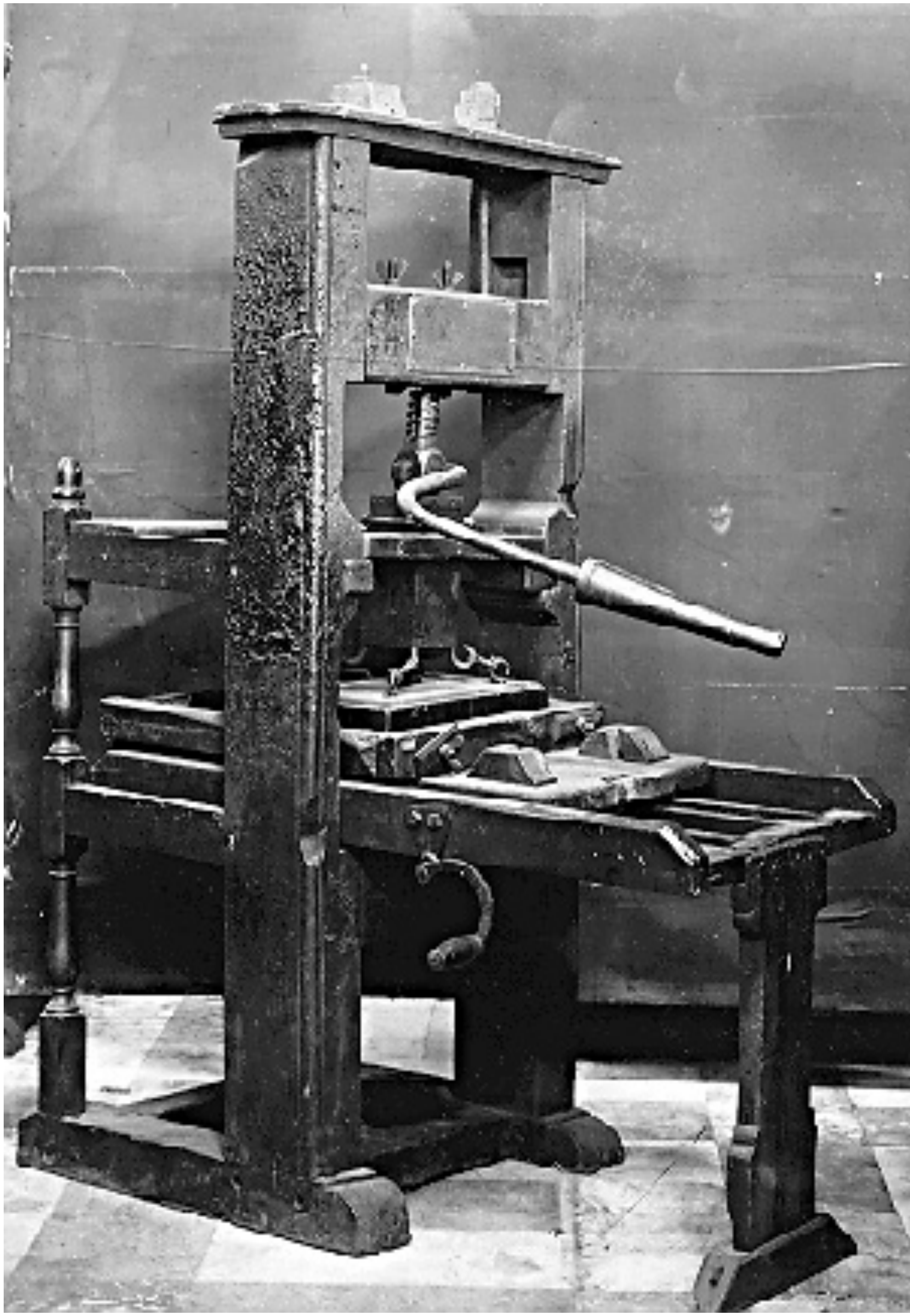
The first real library ~300BC

Discovery of one tradeoff between clay and paper…

You can't have discontinuous reading* until you have a random access technology.

**Indexing and encyclopedias are hard in linear scrolls. Hello ISAM*

Paper Tech v2.1: increased storage density, smaller form factor, durability, high res RGB graphics



Paper Tech v2.2

The change in printing over time accelerates.

Block printing replaced by movable metal type.

The job of production is faster and cheaper.

Commoditization changes the landscape over the next 200 years.

The *printed* becomes more important than the *printer*.

Third Nature

The Elizabethan Era

Production: printing presses

Data management tech:

- Perfect copies
- Topical catalogs
- Font standardization
- Taxonomy ascends

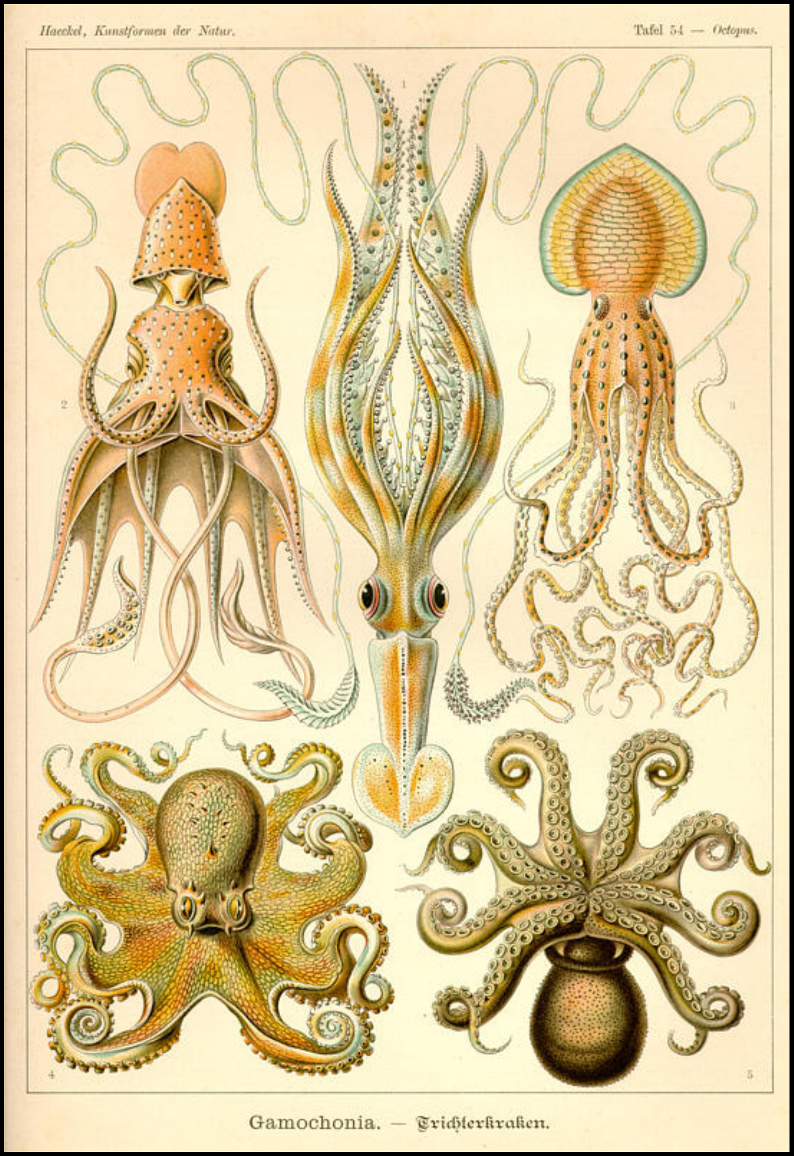
Information explosion:

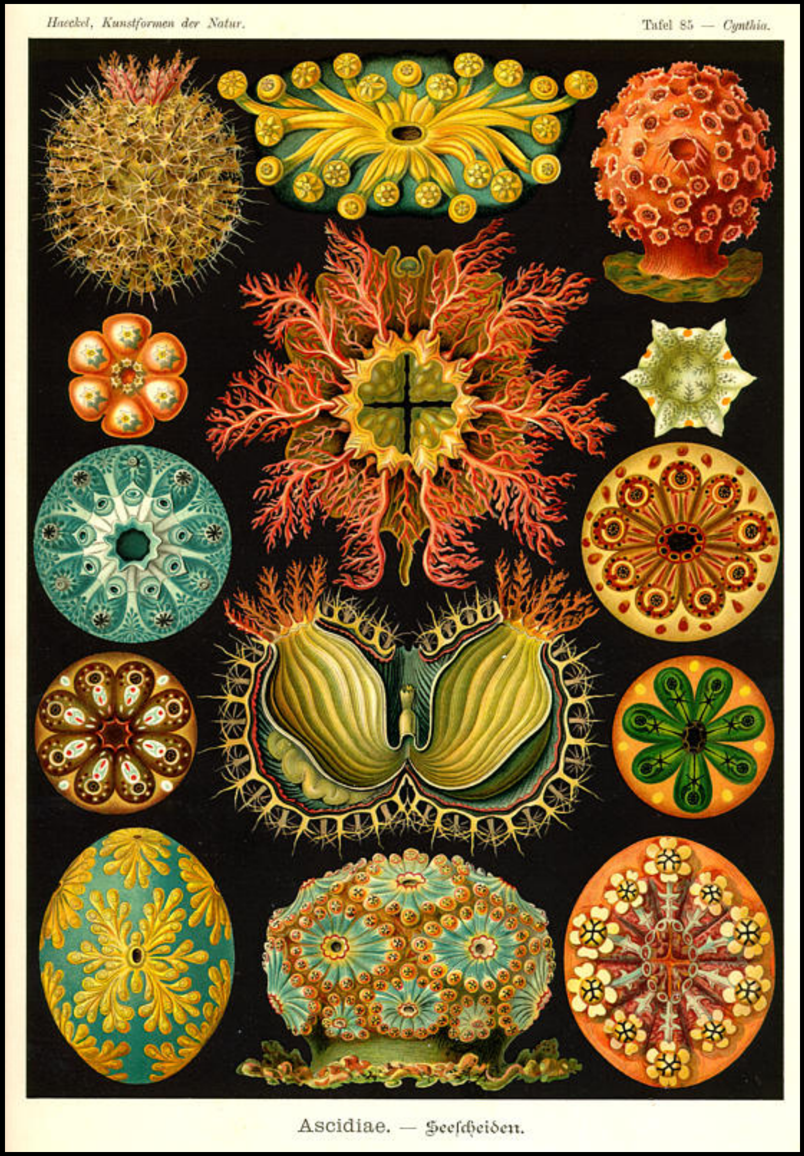
- 8M books in 1500
- 200M by 1600
- Commoditization
- Overload

Third Nature

Better *embedded* metadata: title page, colophon, ToC

The Georgian Era: The Explosion of Natural Philosophy





Sharing knowledge in a larger community required common language, structure

Linnaeus

Top down orientation

Static structure

Descriptive rather than explanatory

Taxonomic classification

Buffon

Bottom up orientation

Flexible structure

Explanatory, descriptive

Faceted classification

SQL

vs

NoSQL

Think about why that happened

Third Nature

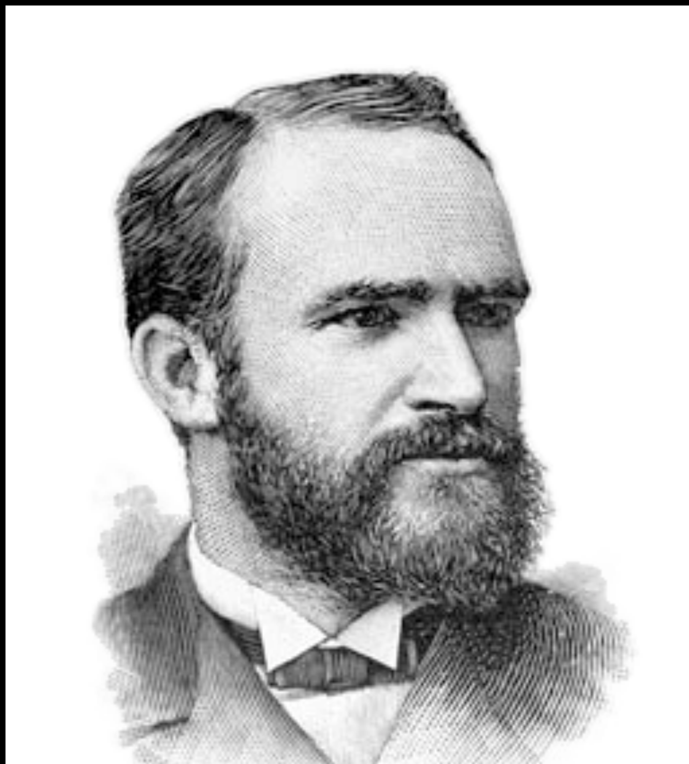
The Victorian Era

The powered printing information explosion:

- Card catalogs, cross-referencing, random access metadata
- Universal classification
- Extended information management debates
- Trading effort and flexibility for storage and retrieval
- <u>Stereotyping</u>

Third Nature

Melvil Dewey



Dewey Decimal System

Top down orientation

Static structure

Descriptive rather than explanatory

Taxonomic classification

Charles Ammi Cutter

Cutter Expansive Classification System (~1882)

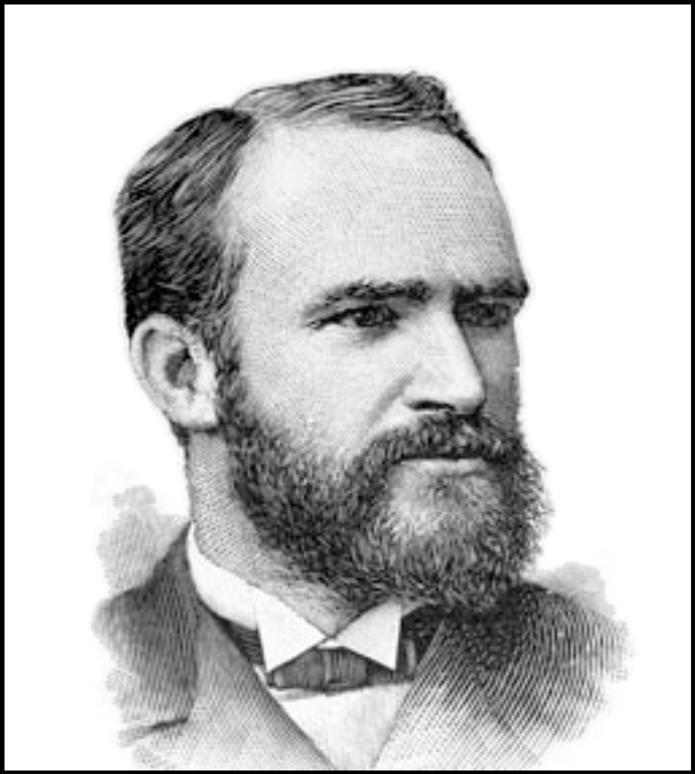
Bottom up orientation

More flexible structure

Explanatory, descriptive

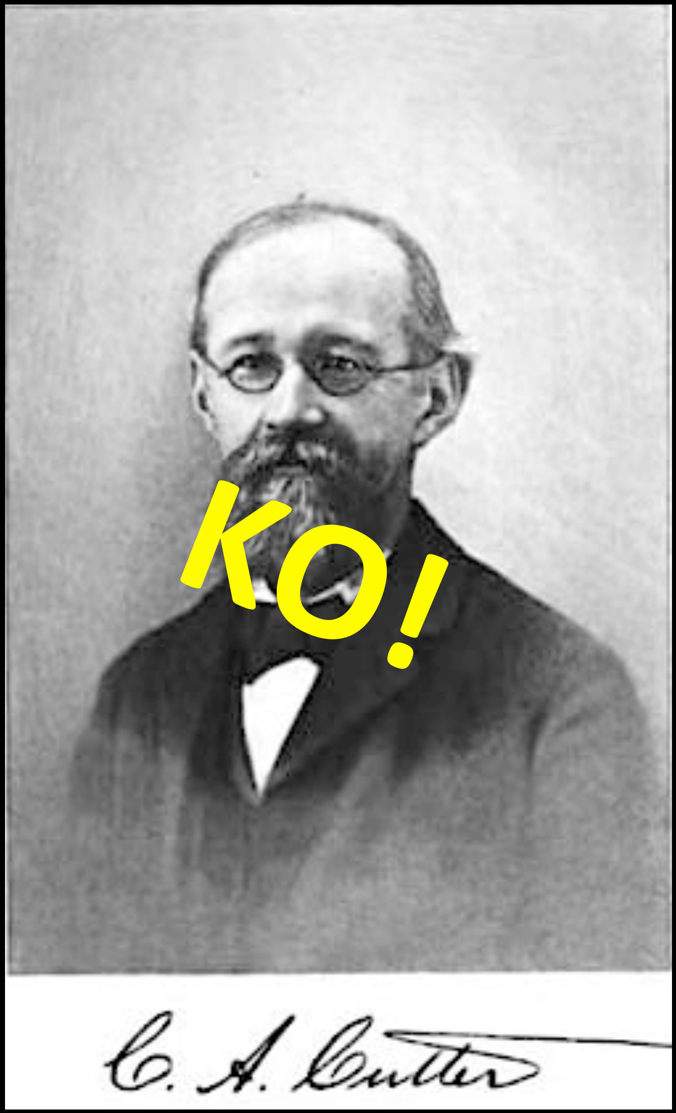
Faceted classification

SQL



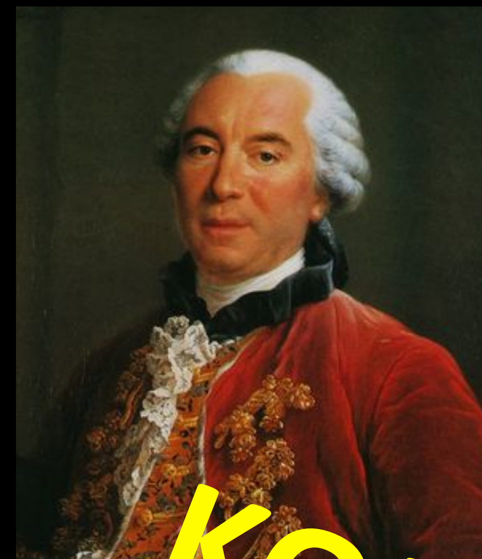
vs

NoSQL



Third Nature

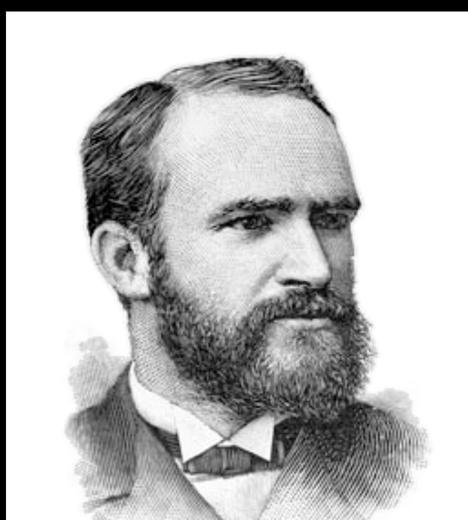
So why did Linnaeus and Dewey win?



KO!

Good enough wins the day

Pragmatism



It wasn't solving the problem you thought it was.

Third Nature

Summarizing

Thousands of years of thought have been put into principles of organization and use. The abstract patterns are the same, only the implementation changed.

- Clay: tablets about tablets, tablets about what's in tablets, 100X increase in data density over counting tech
- Scrolls: scrolls about scrolls, scrolls about what's in scrolls, prepended/appended navigation, >100X increase in density
- Books: books about books, books about what's in books, embedded internal navigation, >1000X increase in density
- Digitized data: similar, far denser, and *different because it isn't locked into physical forms*

Third Nature

History is always the same

Every technology is a trade:

- Top down vs. bottom up
- Authority vs. anarchy
- Bureaucracy vs. autonomy
- Control vs. creativity
- Hierarchy vs. network
- Dynamic vs. static
- Power vs. ease
- Work up front vs. postponed

In every choice, something is lost and something is gained.

Third Nature

What lessons does this history teach us?

1. Information requires organizing principles.
2. Differences in scale require different principles.
3. There are multiple levels of information architecture and principles of organization.
4. At a key point in the adoption cycle, emphasis shifts from collection and management of information to dissemination and use.

First we record, then we use and share.

Like transaction processing, query & analysis.

Third Nature

Information management through human history always follows the same pattern

New technology development

creates

New methods to cope

creates

New information scale and availability

creates…

Third Nature

Big Data

"The most amazing achievement of the computer software industry is its continuing cancellation of the steady and staggering gains made by the computer hardware industry.“ *- Henry Peteroski*

DEALING WITH BIG: SOME SCALING HISTORY

Third Nature

Why doesn't your database scale?

Third Nature

Hipster bullshit

I can't get MySQL to scale

therefore

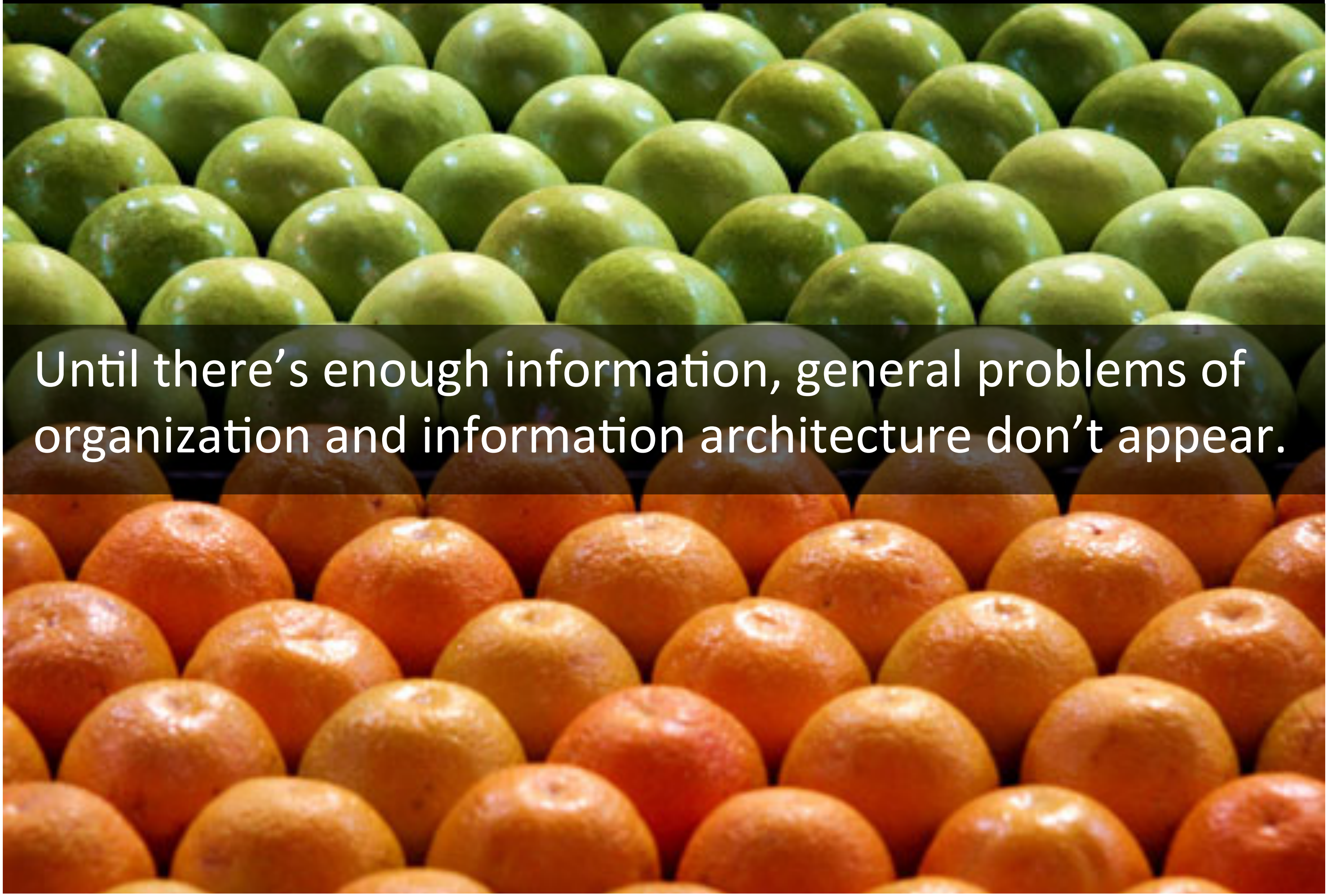
Relational databases don't scale

therefore

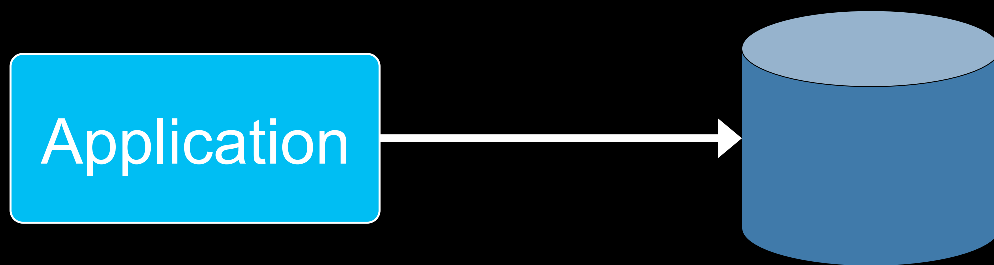
We must use NoSQL* for query too

**including Hadoop and related*

Generation and collection always come first

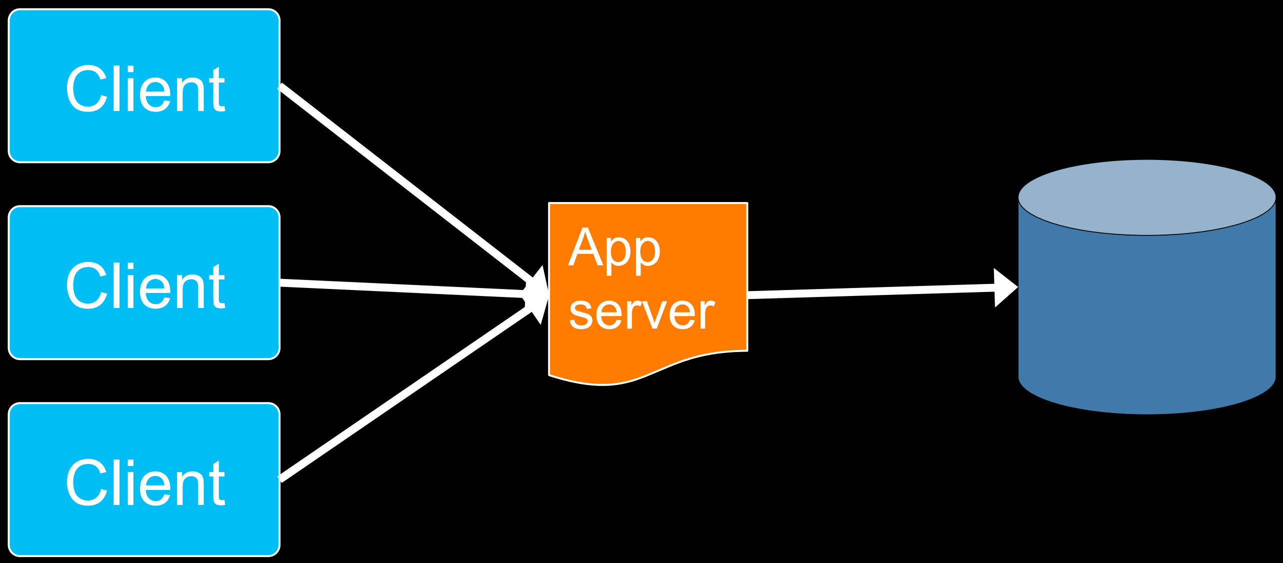


The early days: client/server as the starting point



We had transaction processing against the DB all on the same machine. Then on two separate machines.

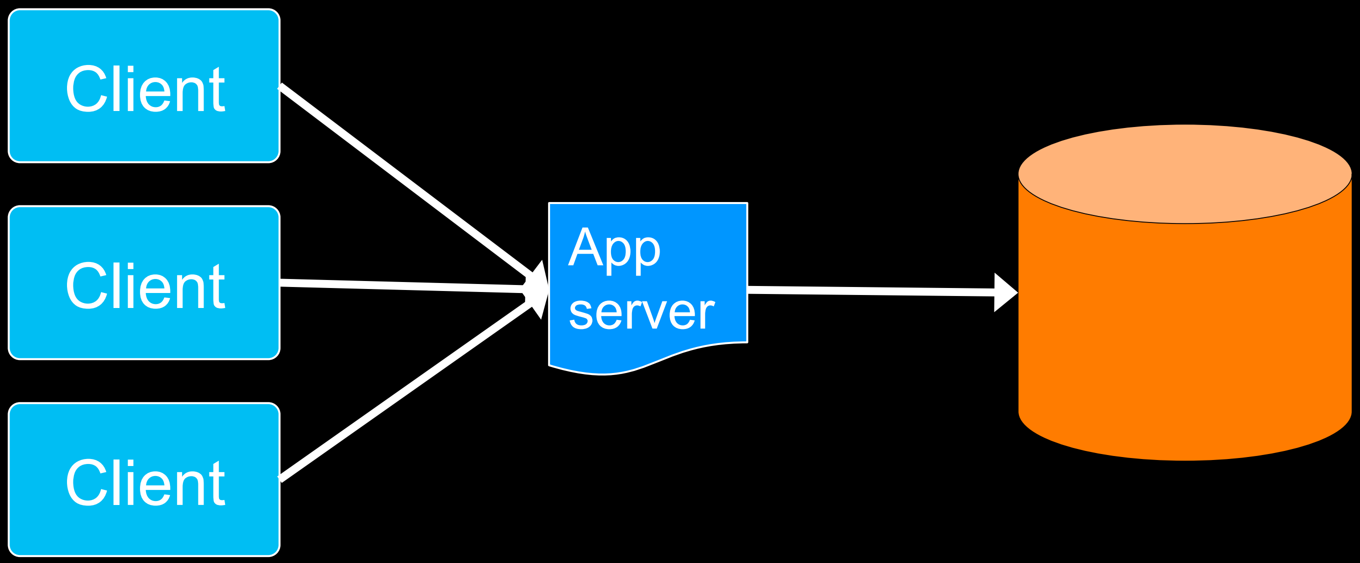
Scaling client/server



We added app servers and pooled connections.

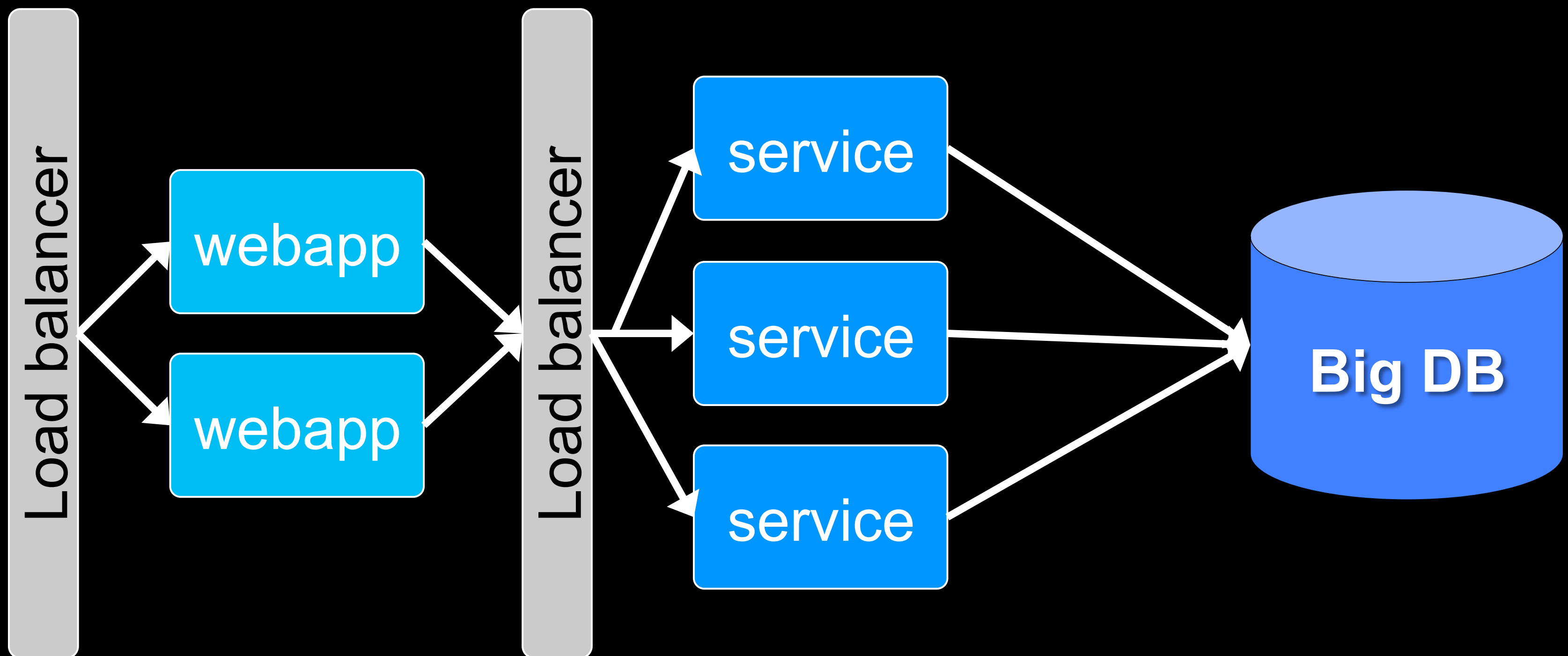
Third Nature

Scaling client/server



Then threw money at the problem in the form of hardware (made the database bigger).

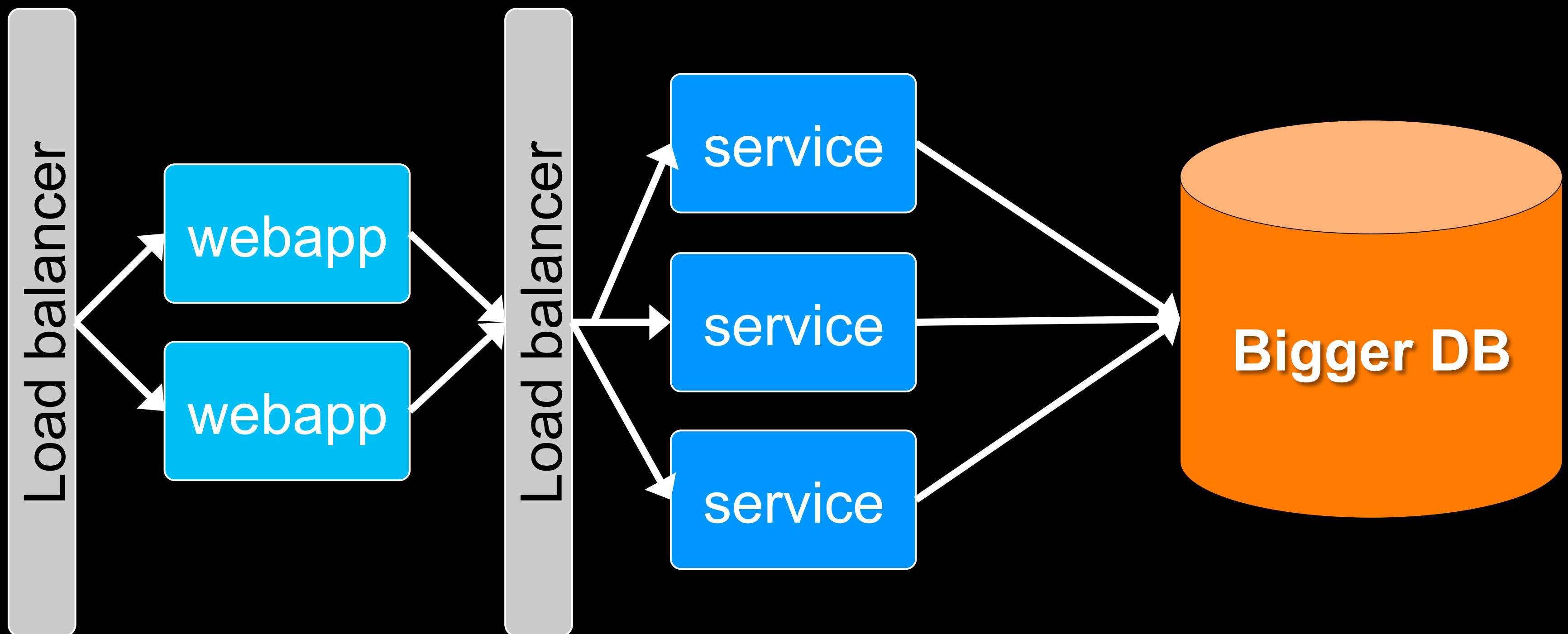
Web apps were a huge increase in concurrency



Architecture changed to reflect new stateless model.

We had scalability and availability problems.

Increasing traffic?

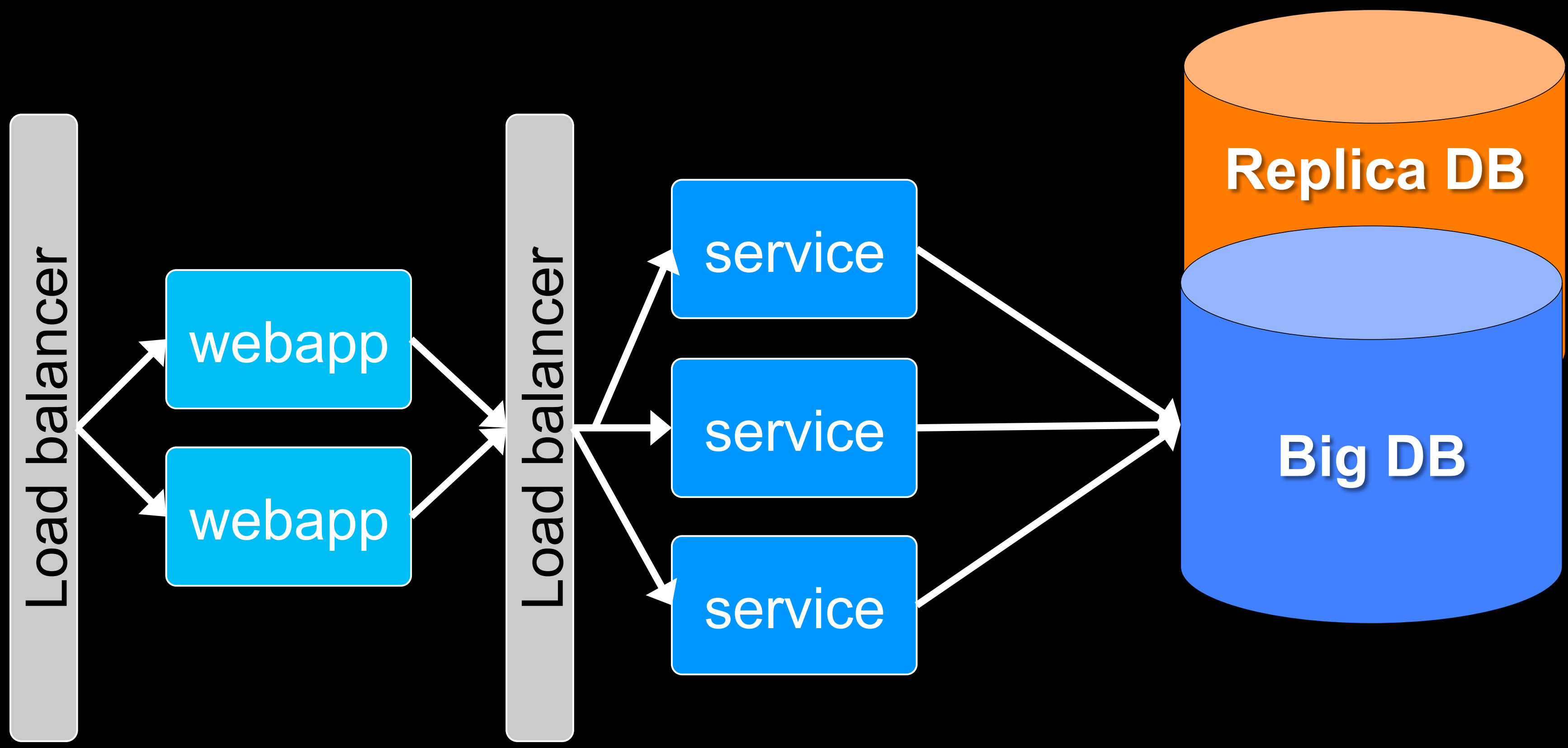


Keep adding hardware, make the DB bigger.

Limits reached, performance, scalability and availability problems.

Third Nature

Increasing traffic?

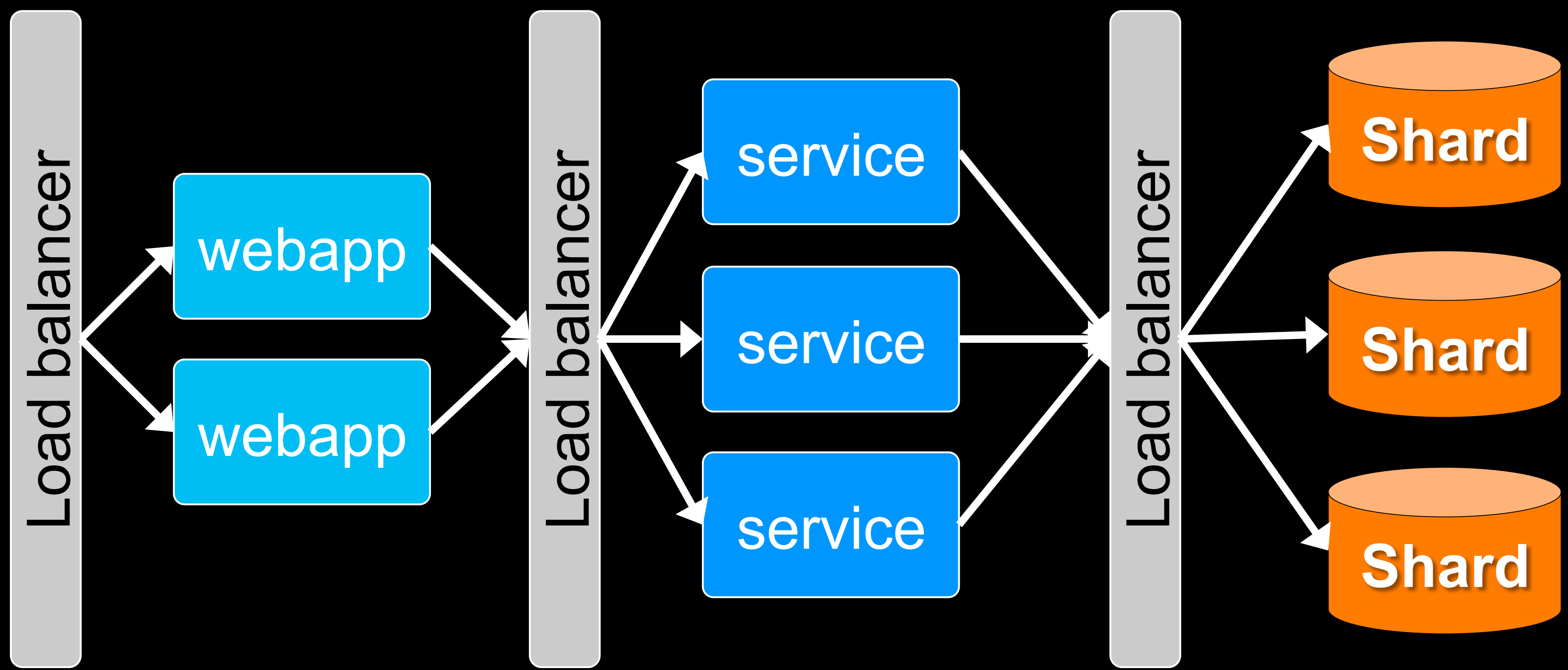


Read-only replicas will save the day!

Still have scalability and availability problems.

And now operational overhead and problems.

Increasing traffic?

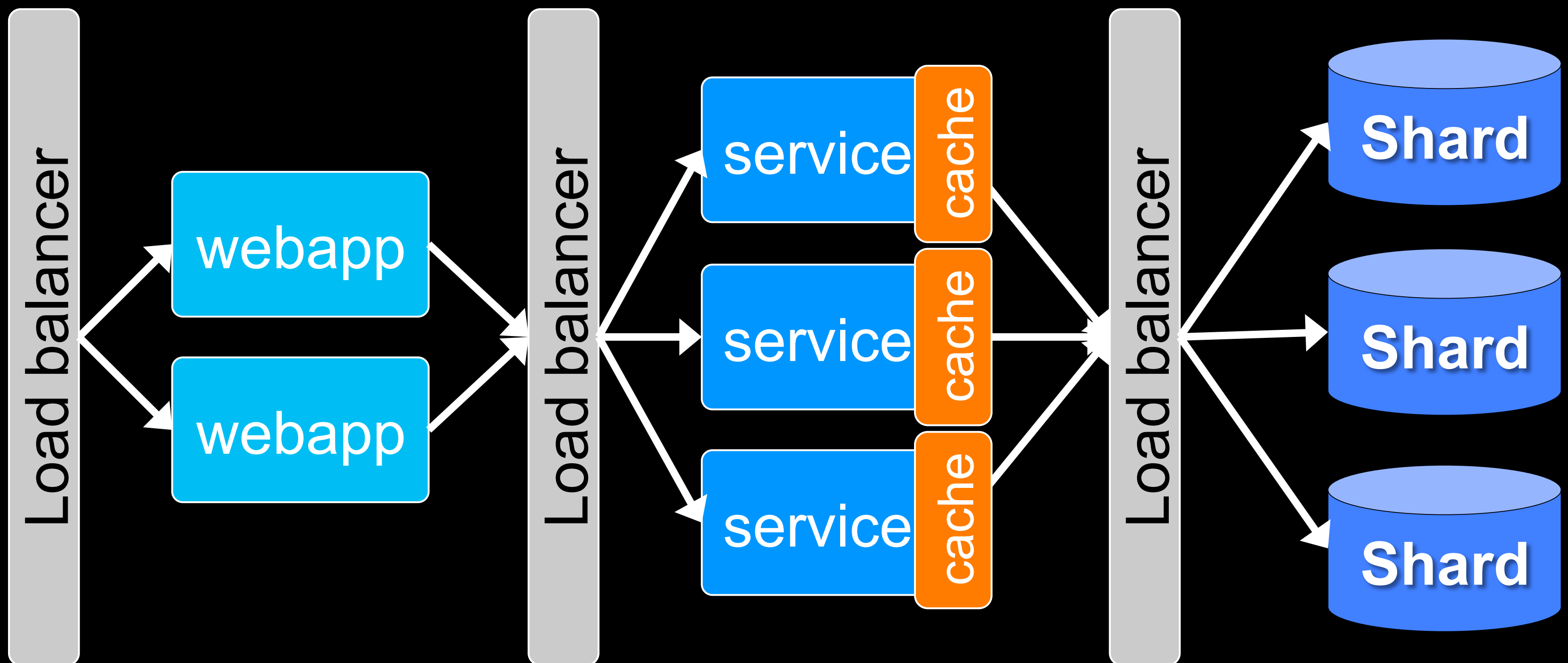


Sharding seems a fine thing. But it's one letter from…

Scaling and perf better, overhead and operational complexity high and worsening.

Third Nature

Increasing traffic?



Let's cache data at the service tier!

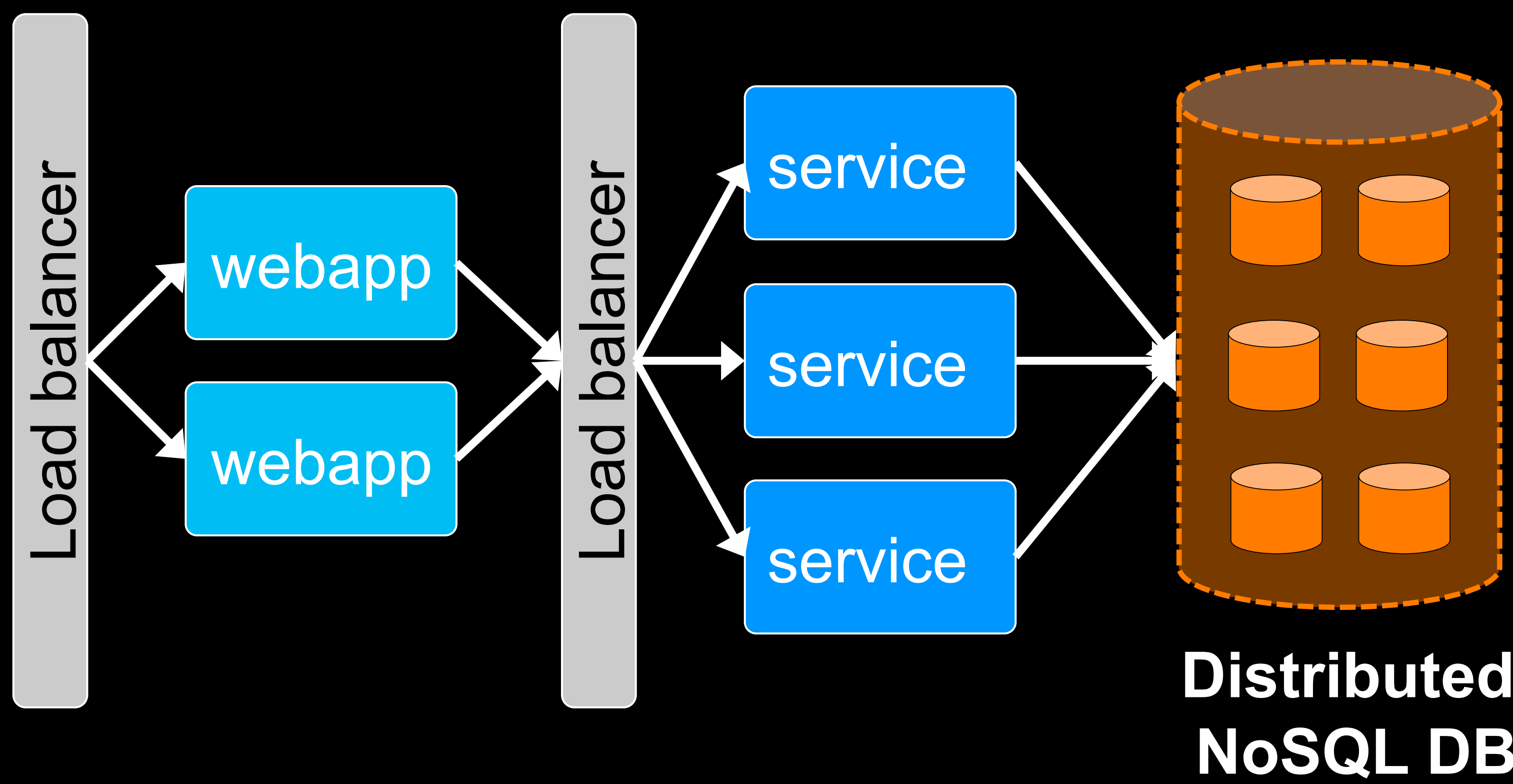
Performance better, overhead and operational complexity higher.

Third Nature

What are the problems now?

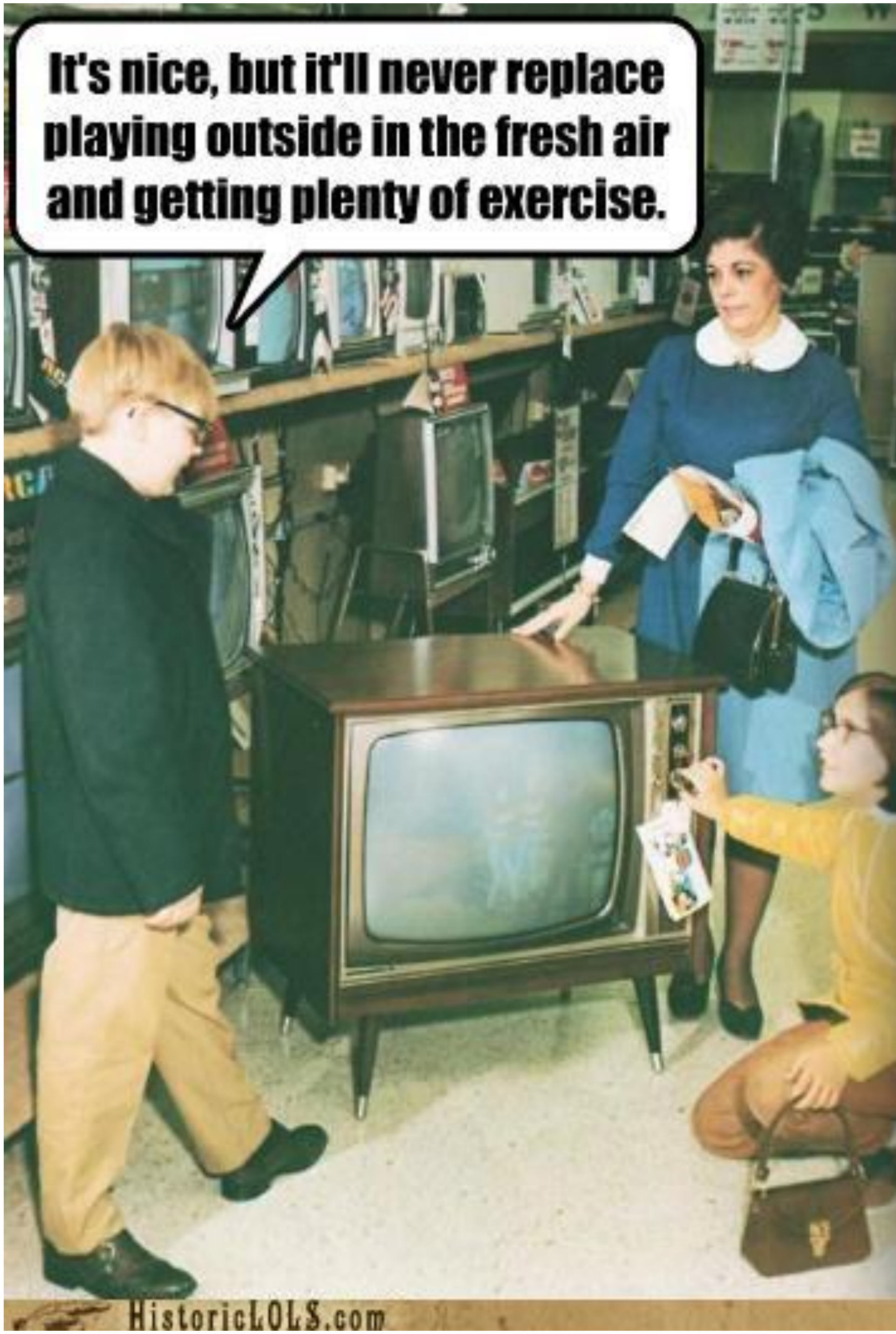
1. More hardware, more things to break
2. More management and administration
3. More software complexity
4. Increasing distance for data to travel = latency
5. Data administration difficult to impossible

Problem solved?



Distributed NoSQL DB (handles cache, load balance, data distribution). Similar performance, simpler scaling, reduced operational problems, simpler application architecture. Finished!

Third Nature



TANSTAAFL

When replacing the old with the new (or ignoring the new over the old) you always make tradeoffs, and usually you won't see them for a long time.

Technologies are not perfect replacements for one another. Often not better, only different.

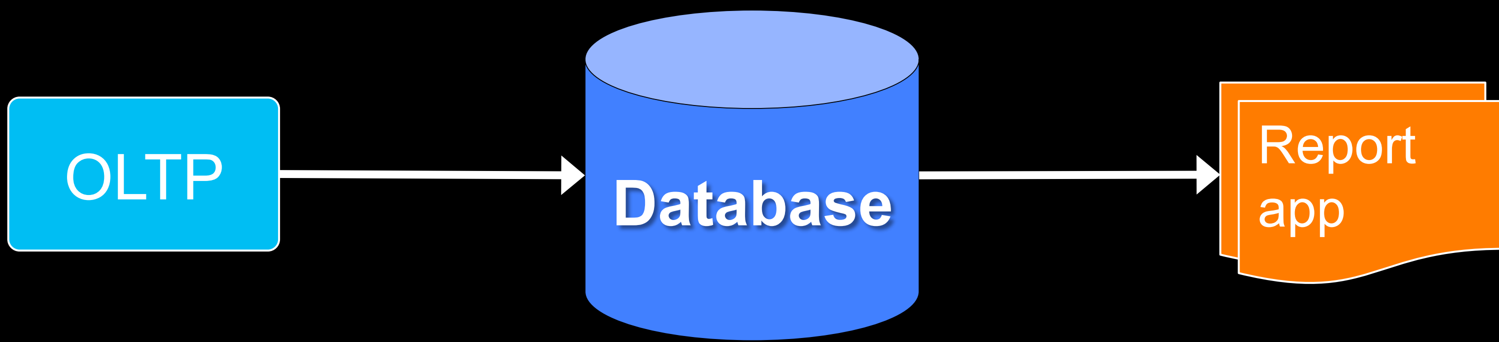
Third Nature

Not finished: remember the cycle of history…

The biggest hole in the prior section on scaling is that we scaled OLTP, what about OLAP?

Queries <> transactions.

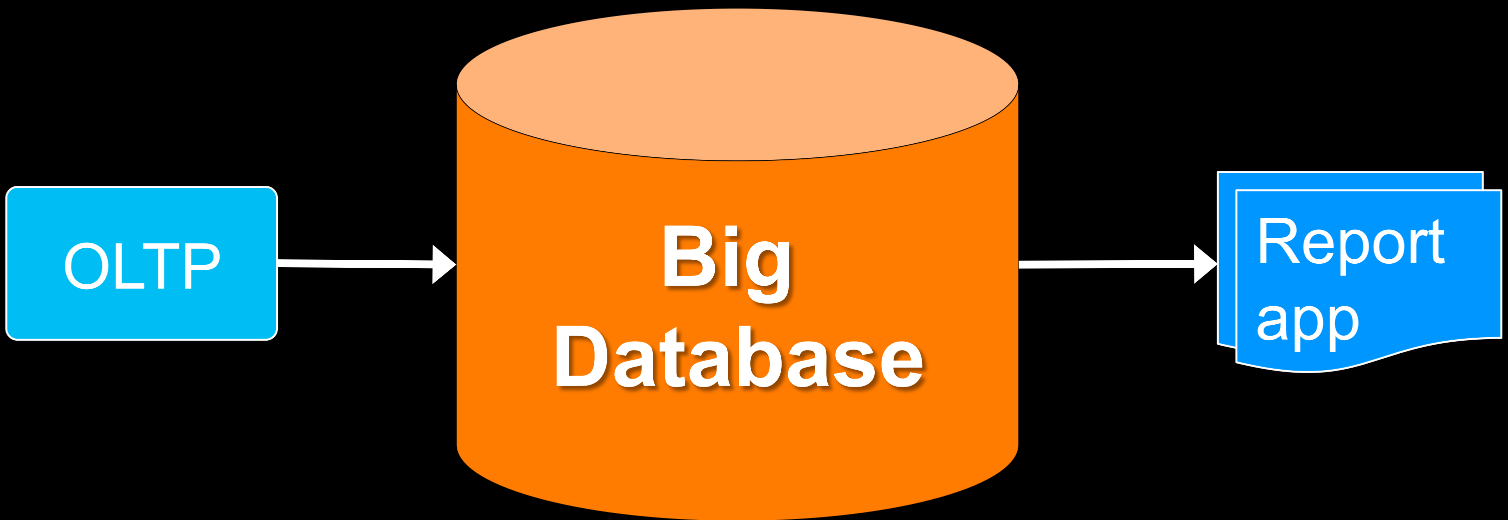
Solving query problems



Aggregate or low selectivity queries were a problem early on, when people wanted to *use* the data.

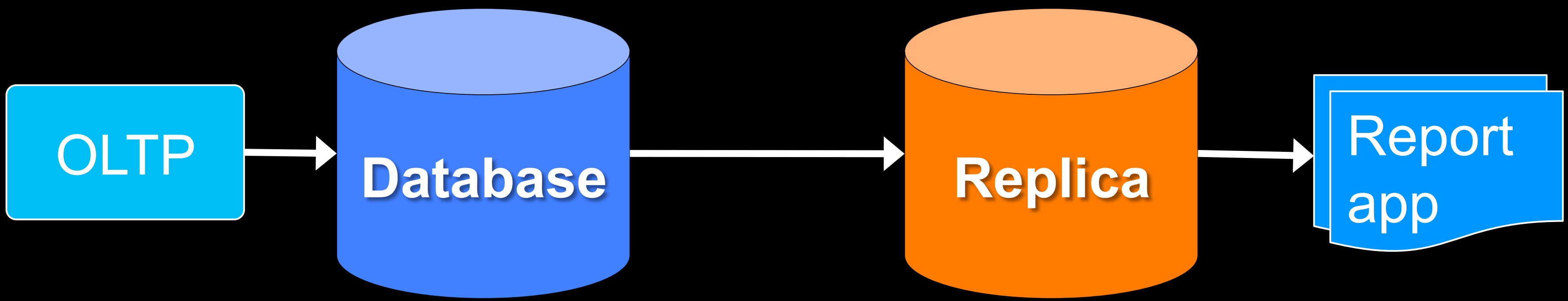
Every report or query is a program.

Increasing data volume



Make it faster by throwing money at hardware (sound familiar?)

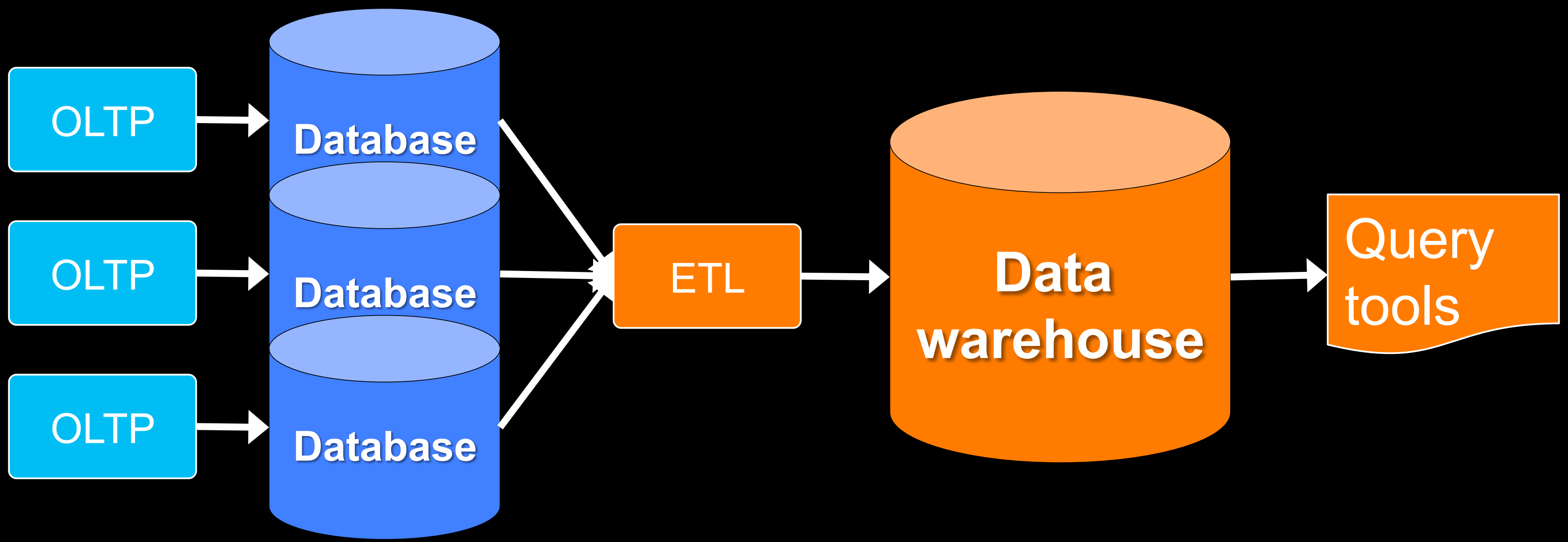
Increasing data volume



Replicas: split the workload and tune the systems based on their workload.

Third Nature

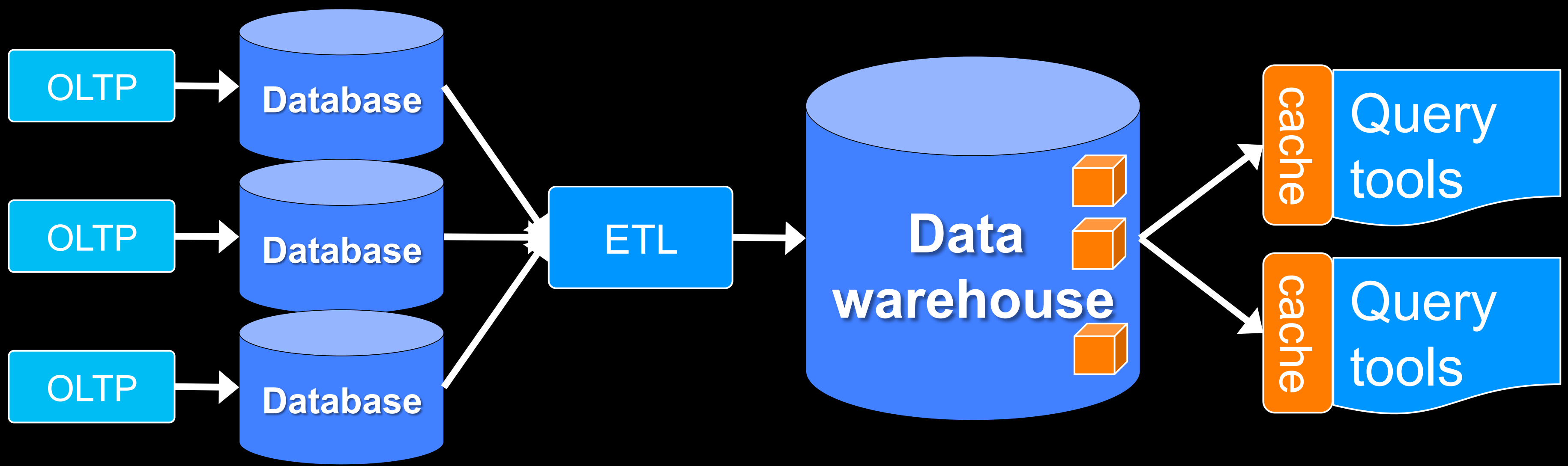
Increasing data volume breaks the old model



Devise a new architecture.

Reschematize the database, eliminate cyclic joins, selective denormalization, *query generators*. But it takes bulk processing to reschematize the data.

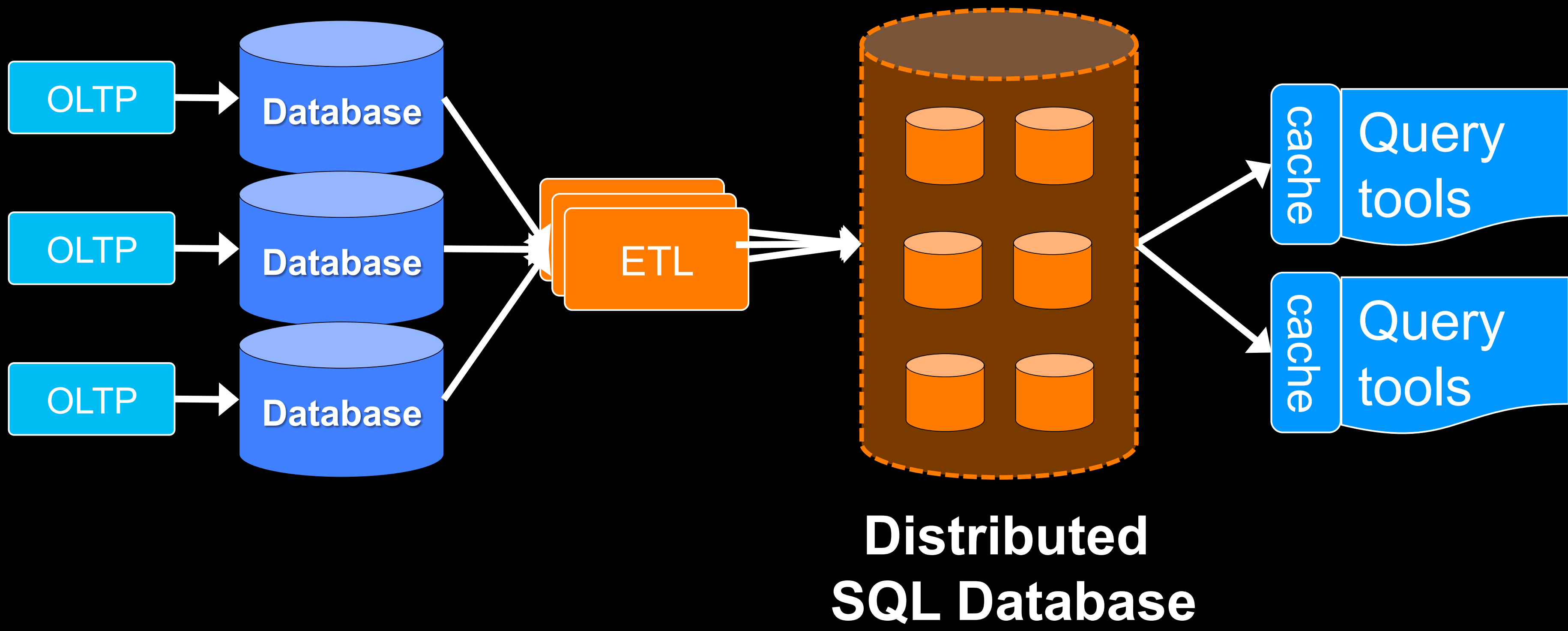
Increasing data volume



Improve response time with caching in the query tools, and by using MOLAP tools that map into cache or memory. Like mainframe reporting. Or…

Third Nature

Increasing data volume



Parallel processing for ETL. Distributed query databases for fine grained high volume parallelism.

Third Nature

The architecture looks familiar

Two workloads, two not dissimilar architectures:

- Load-balanced front ends
- Distributed caching layers
- Scalable distributed parallel databases

But the nature of the OLTP and OLAP workloads is very different. Forcing them into one platform is almost impossible for data architecture reasons and particularly at scale*

Why would digital data be any different than clay or scrolls or books?

DATA PERSISTENCE AND STORES

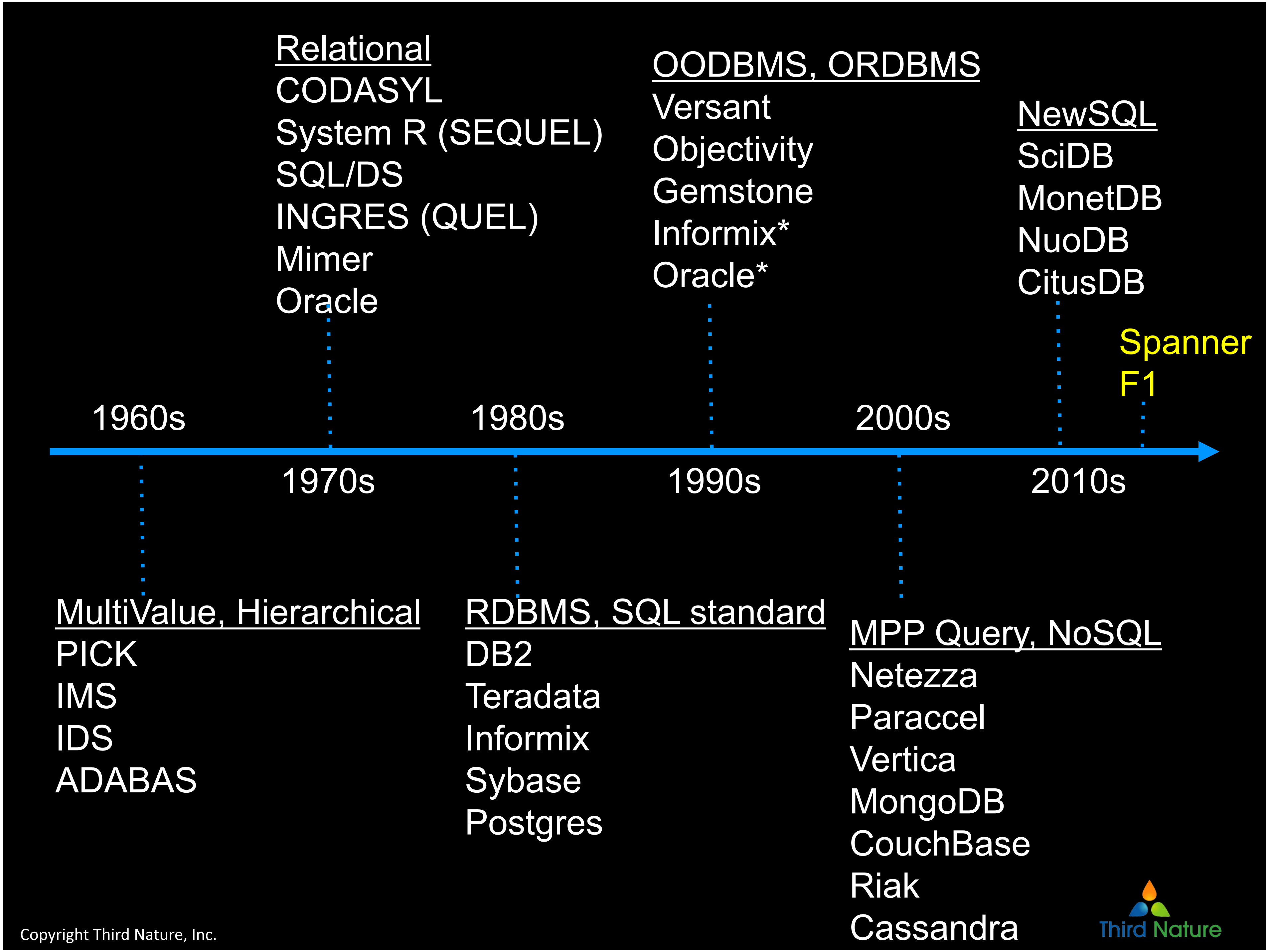
Third Nature

"Big data is unprecedented."

- Anyone involved with big data in even the most barely perceptible way

There's a difference between having no past and actively rejecting it.

Third Nature


Relational
CODASYL
System R (SEQUEL)
SQL/DS
INGRES (QUEL)
Mimer
Oracle
OODBMS, ORDBMS
Versant
Objectivity
Gemstone
Informix*
Oracle*
NewSQL
SciDB
MonetDB
NuoDB
CitusDB
Spanner
F1
1960s
1970s
1980s
1990s
2000s
2010s
MultiValue, Hierarchical
PICK
IMS
IDS
ADABAS
RDBMS, SQL standard
DB2
Teradata
Informix
Sybase
Postgres
MPP Query, NoSQL
Netezza
Paraccel
Vertica
MongoDB
CouchBase
Riak
Cassandra
Copyright Third Nature, Inc.
Third Nature

A history of databases in No-tation

1970: NoSQL = We have no SQL

1980: NoSQL = Know SQL

2000: NoSQL = No SQL!

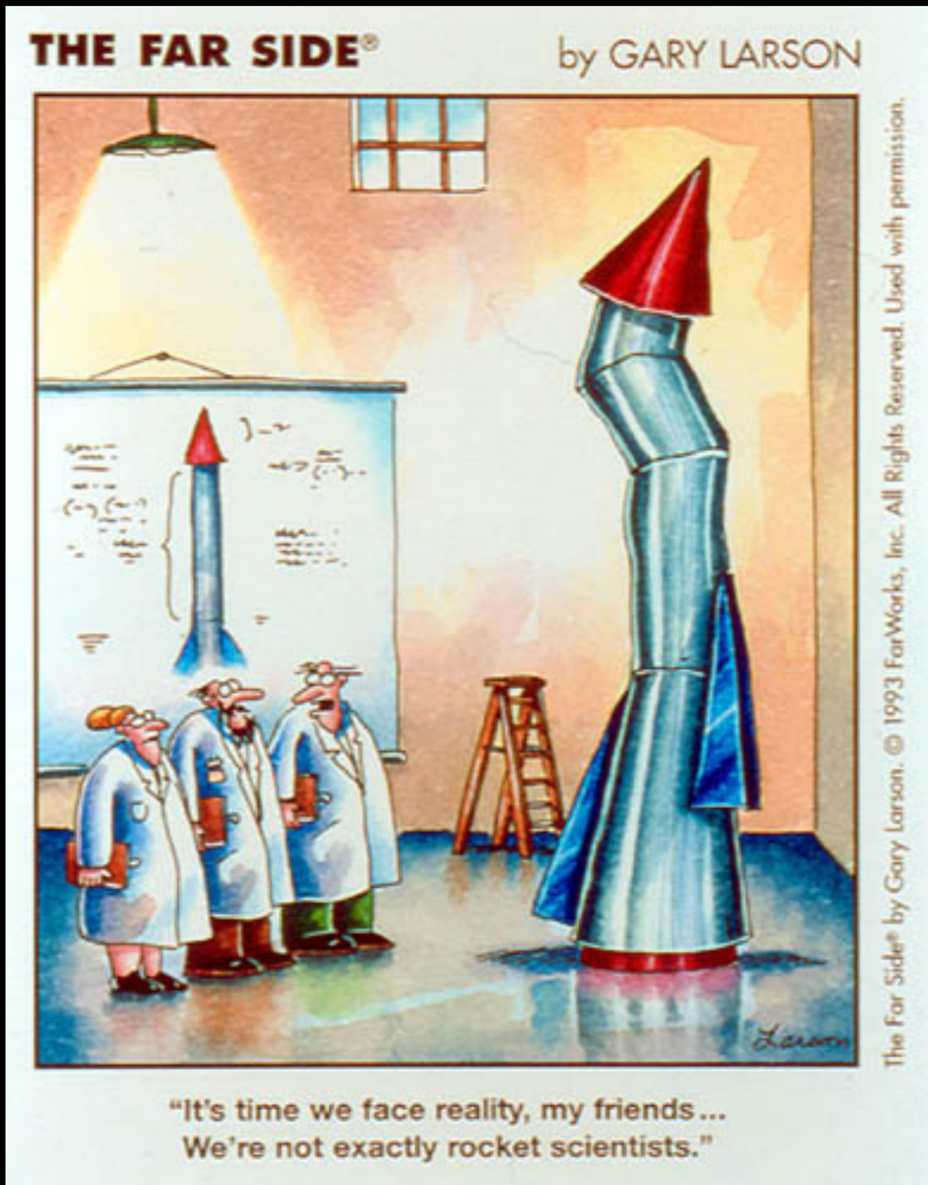
2005: NoSQL = Not only SQL

2013: NoSQL = No, SQL!

(R)DB(MS)

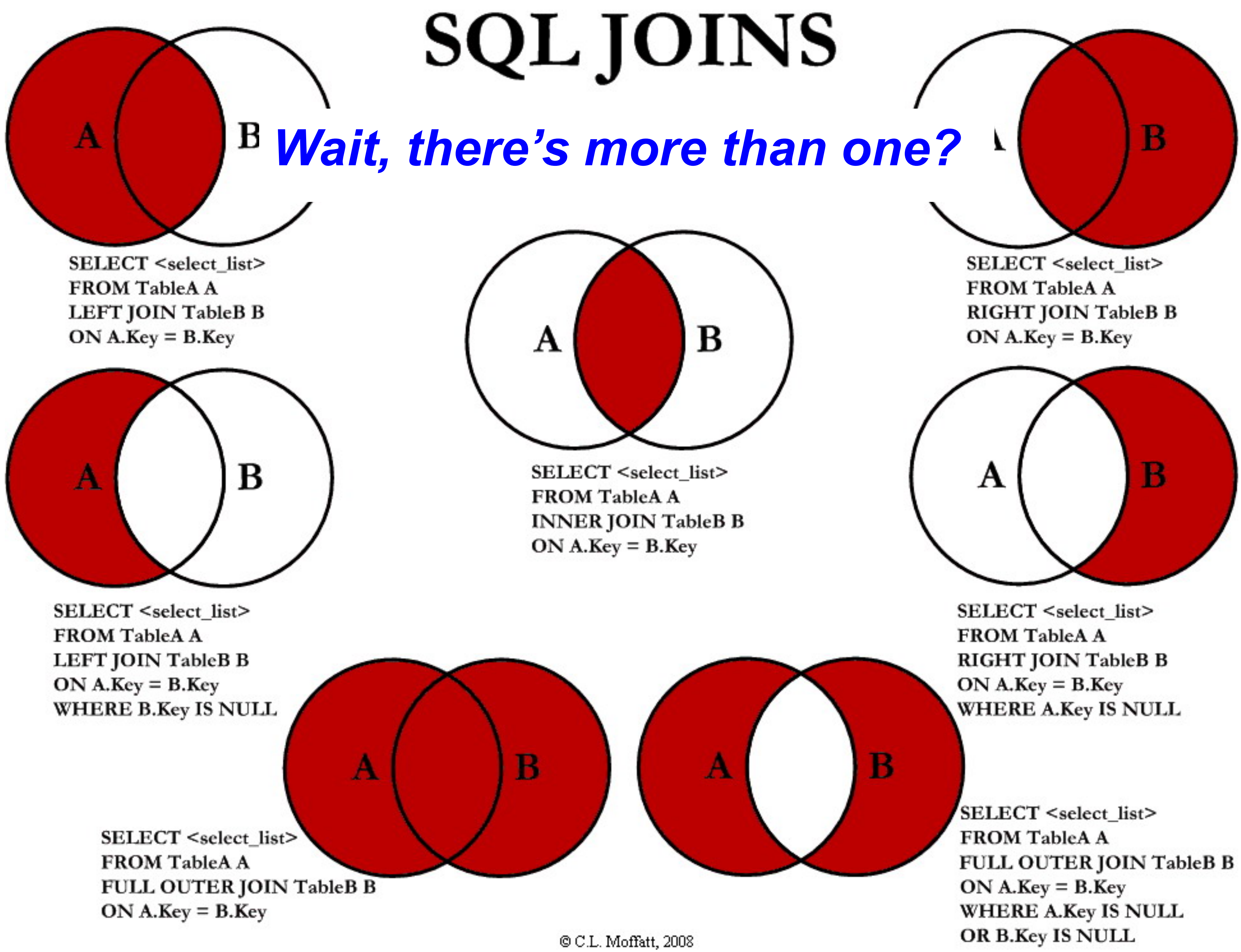
Third Nature

Tradeoffs?



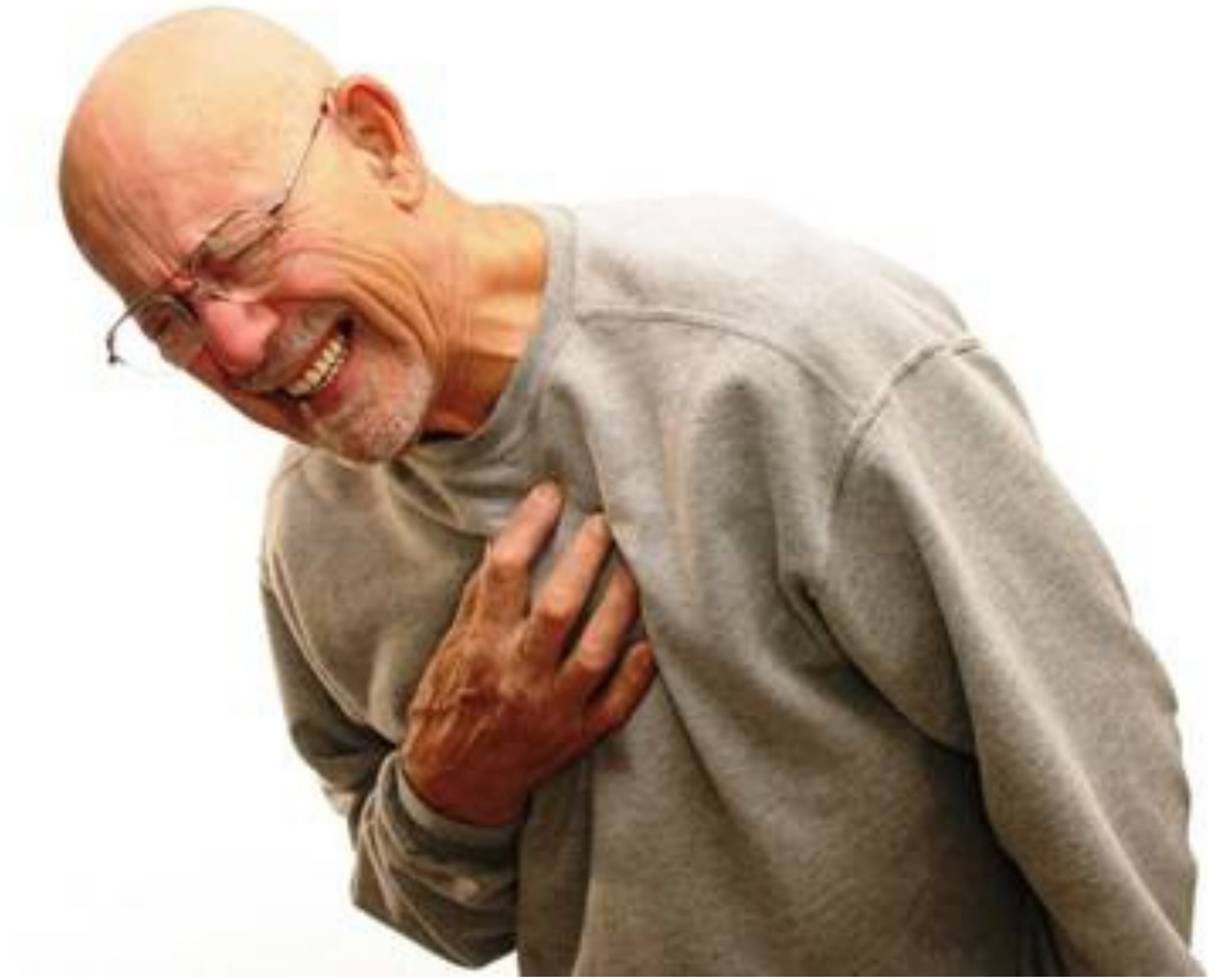
"Query optimization is not rocket science. When you flunk out of query optimization, we make you go build rockets."

Third Nature



Wait, there's more than one?

In NoSQL Land, Optimizer is You!



Third Nature

Tradeoffs: In NoSQL, the DBMS is You Too

SQL database

Application

Database

Services provided

Standard API/query layer*

Transaction / consistency

Query optimization

Data navigation, joins

Data access

Storage management

NoSQL database

Application

Database

Anything not done by the DB becomes a developer's task.

Third Nature

Relational: a good conceptual model, but a prematurely standardized implementation

The relational database is the franchise technology for storing and retrieving data, but…

1. Global, static schema model
2. No rich typing system
3. No management of natural ordering in data
4. Many are not a good fit for network parallel computing, aka cloud
5. Limited API in atomic SQL statement syntax & simple result set return
6. Poor developer support (in languages, in IDEs, in processes)

Third Nature

Relational: a good conceptual model, but a prematurely standardized implementation

What I did not list:
Scalability and performance

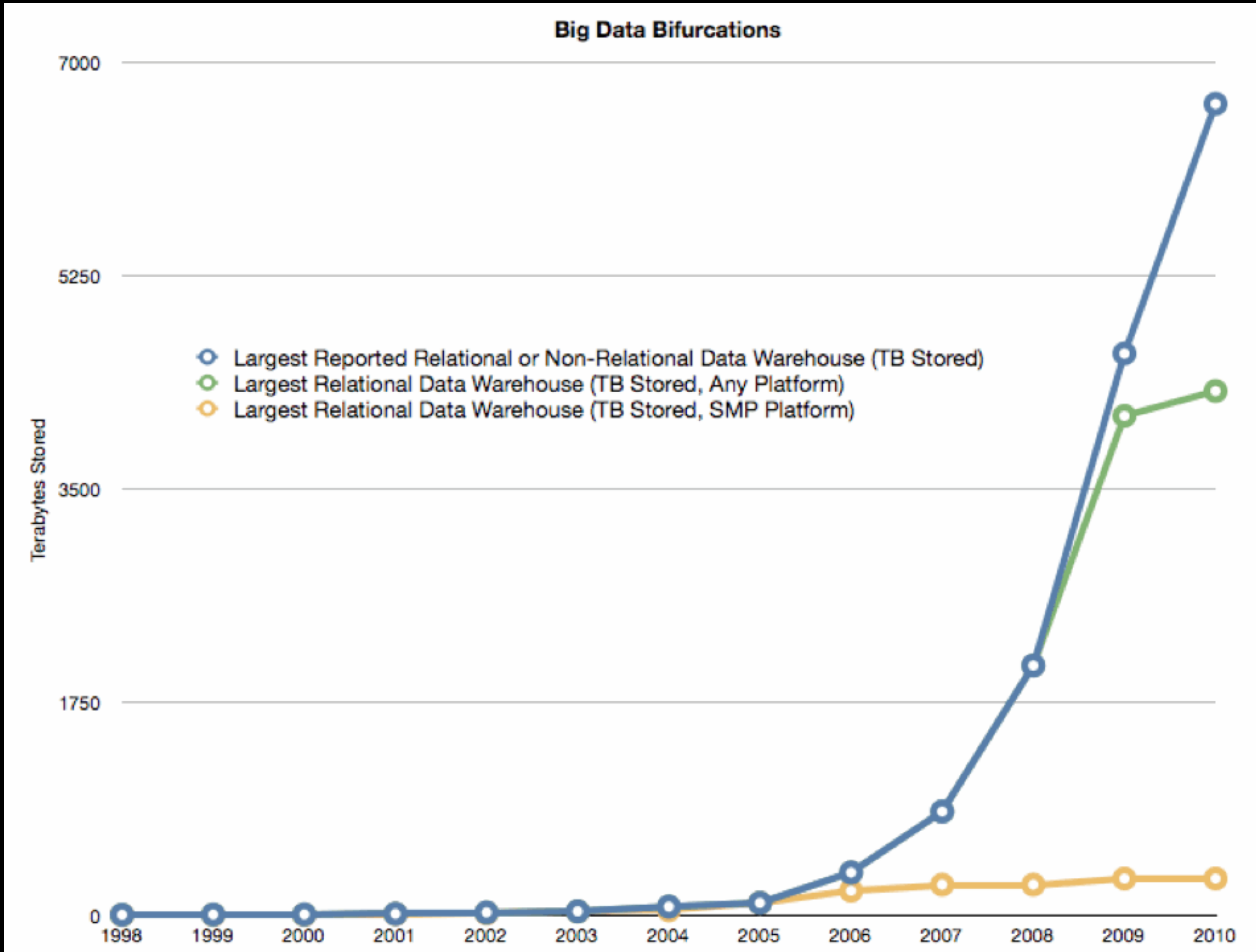
The relational database is the franchise technology for storing and retrieving data, but…

1. Global, static schema model
2. No rich typing system
3. No management of natural ordering in data
4. Many are not a good fit for network parallel computing, aka cloud
5. Limited API in atomic SQL statement syntax & simple result set return
6. Poor developer support

Third Nature

BIGNESS AND SCALABILITY

Technology Capability and Data Volume



Source: Noumenal, Inc.

Third Nature

You can make a database emulate a KVS

If you map the shared event fields to fixed columns and an event type, then use a varchar or clob payload column, you can store arbitrary events in a database and query them via SQL and views (or column functions), and do it all in a single table.

Date	IP	App	EventType	Payload
11/30/11	192.0.168.1	myapp	Event-1	f63jdk5tek8367

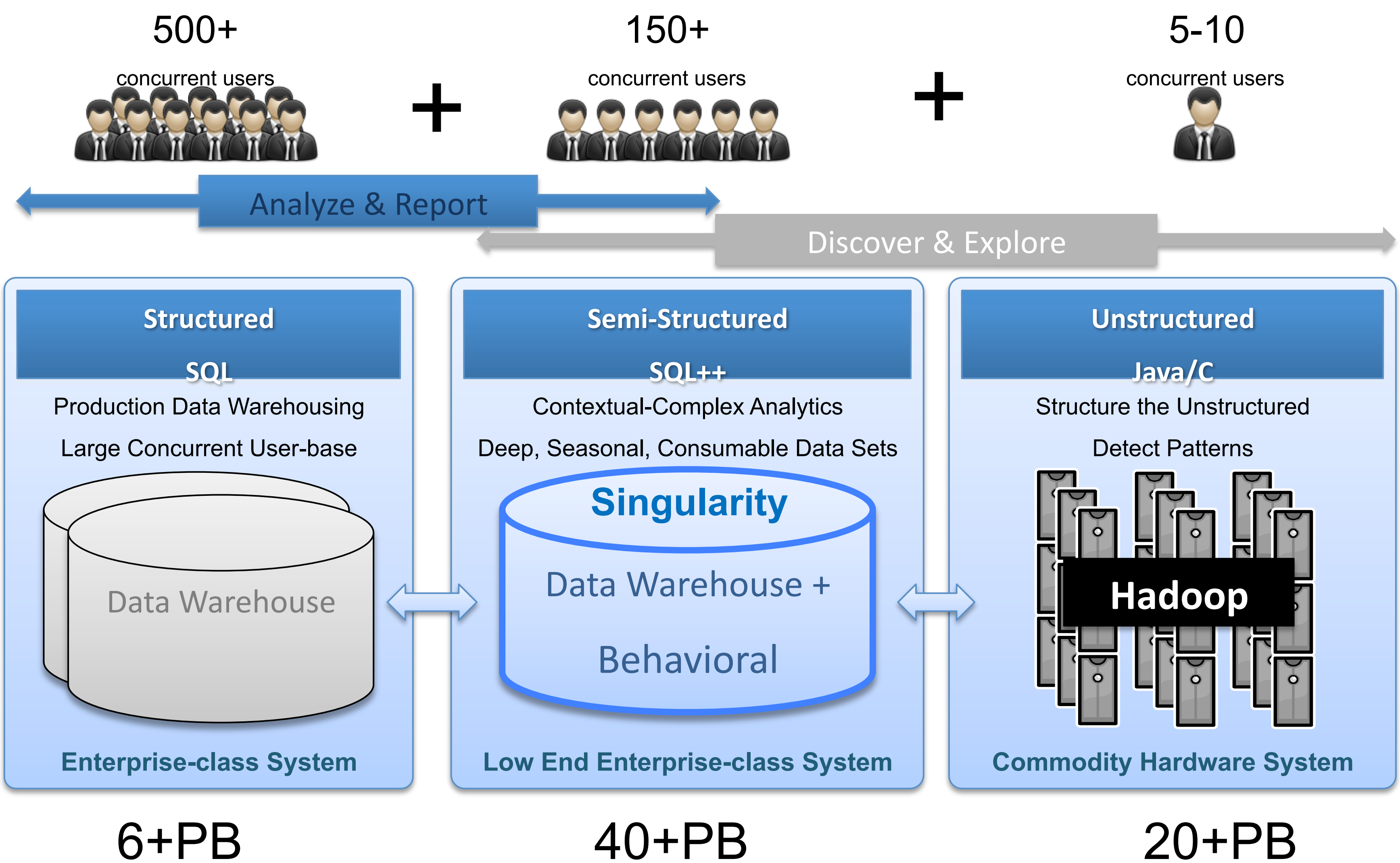
Data common to all events in the database

Type code used to differentiate event payload formats.

Arbitrary data parsed at query time using native DB features like regex or UDF

Third Nature

Data Platforms



Thanks to eBay for these case slides.

How to use a database for semi-structured data

First the user-defined function (UDF)

Start_dt	Guid	Sess_id	Page_id	Soj
2011-10-18	1234	1	15	Language=en& source=hp& itm=i1,i2,i3,i4,i5

```
SELECT start_dt, guid, sess_id, page_id,
            NVL(e.soj, 'itm') AS item_list
FROM    event e
WHERE e.start_dt = '2011-10-18'
            AND e.page_id = 3286                /* Search Results */
```

Start_dt	Guid	Sess_id	Page_id	Item_list
2011-10-18	1234	1	15	i1,i2,i3,i4,i5

Thanks to eBay for these case slides.

How to use a database for semi-structured data

Then the table function (standard ANSI SQL)

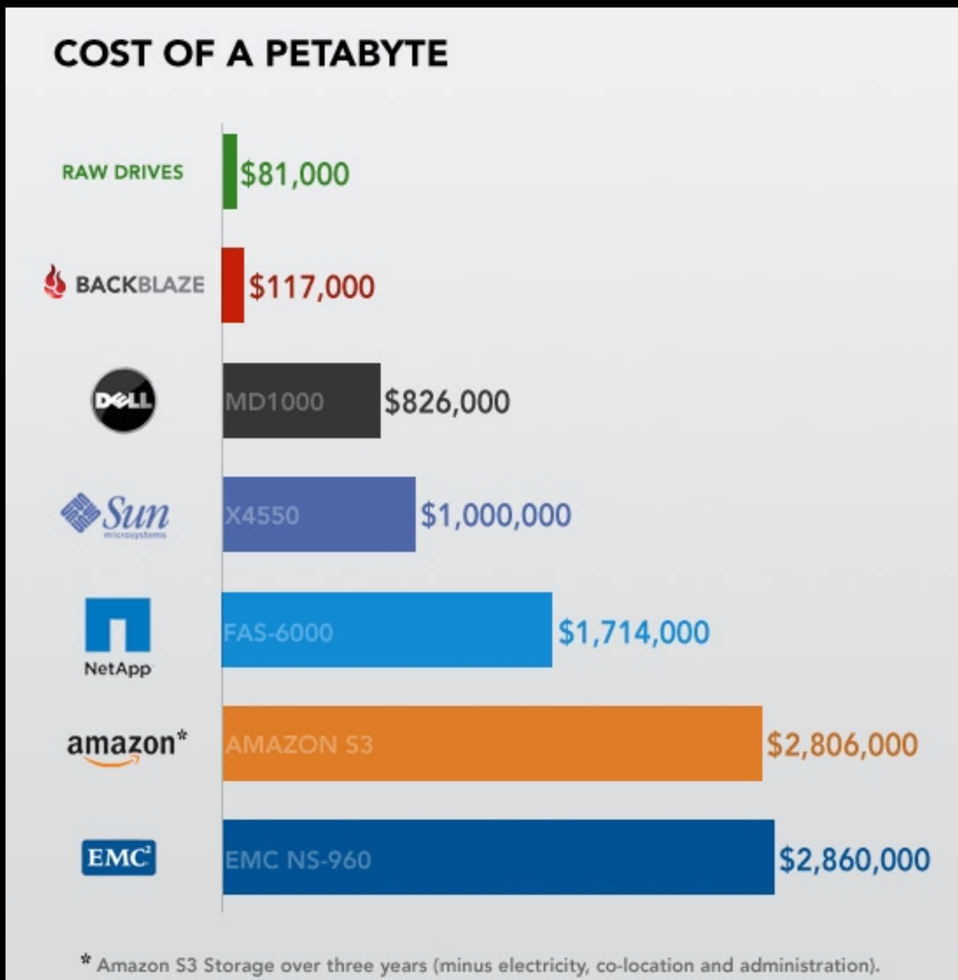
```
WITH event (start_dt, item_list) AS (<previous SQL>)
SELECT
        start_dt,
        item_id,        /* Individual Item */
        count(*)
FROM    TABLE ( /* Normalize comma delimited list */
        normalize_list( start_dt, item_list, ‘,’)
        RETURNS(start_dt, idx, item_id)
        )
GROUP BY 1, 2
ORDER BY 3 DESC
```

*syntax simplified

Start_dt	Item_id	Count(*)
2011-10-18	i1	555
2011-10-18	i2	444
2011-10-18	i3	333
2011-10-18	i4	222
2011-10-18	i5	111

Thanks to eBay for these case slides.

Pricing and performance: Hadoop is a storage and processing play, *not* a database play*



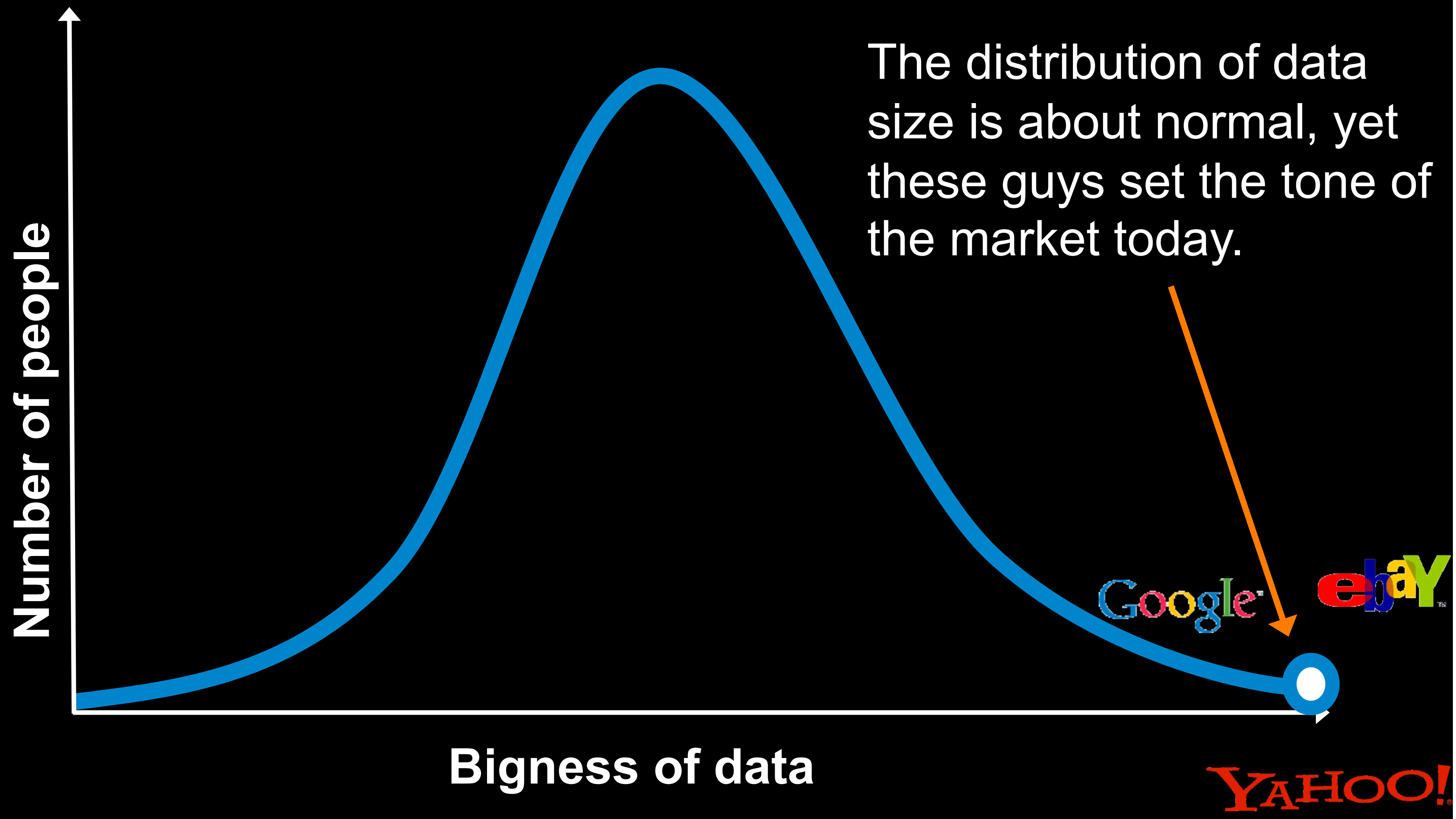
Source: Venturebeat

With big data systems, the cost of storing data is an order of magnitude lower than with databases today (but not the cost or ability to query it back out).

Processing data at scale is at least an order of magnitude cheaper too.

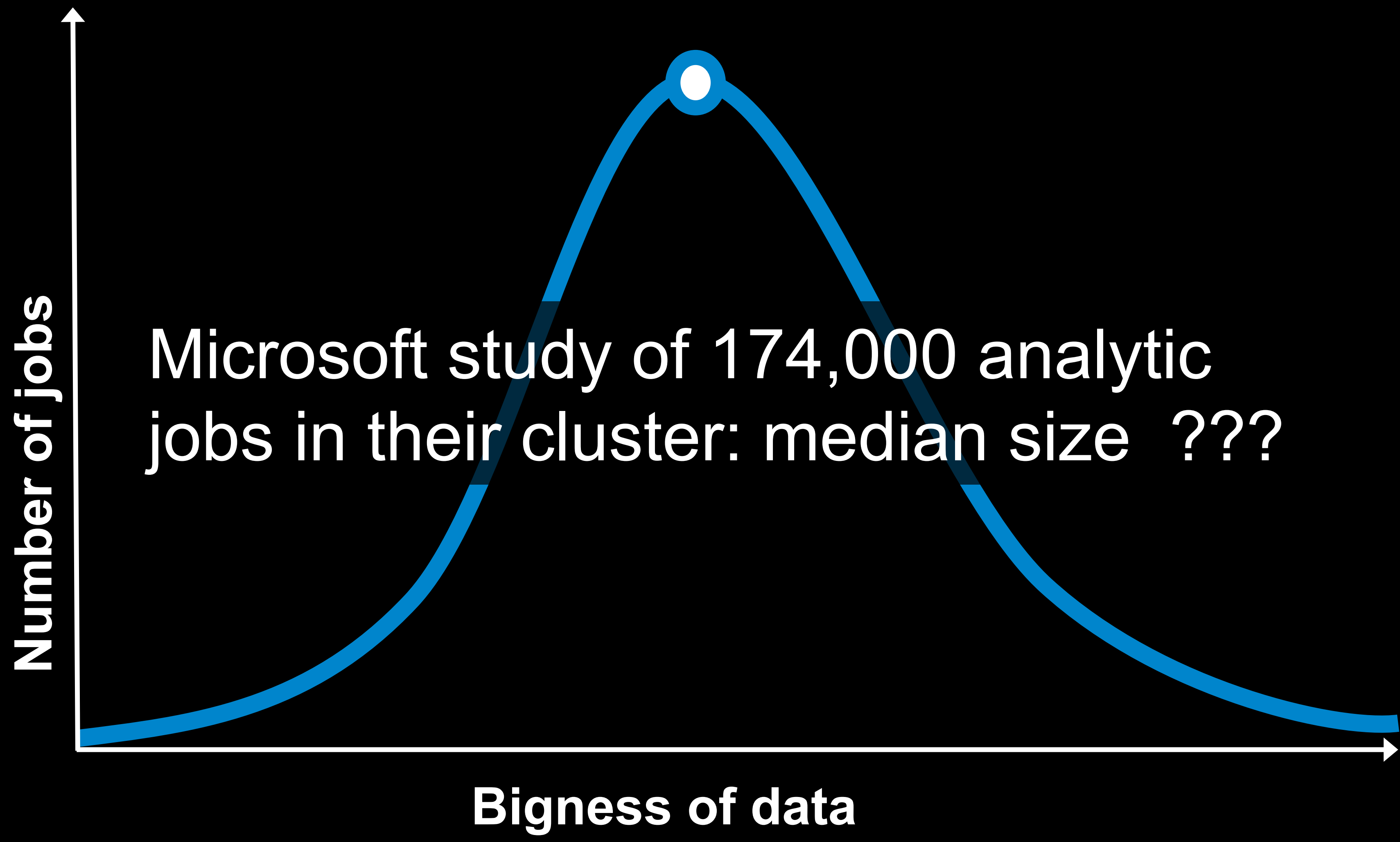
Third Nature

Bigness: most people do not need special technology



Third Nature

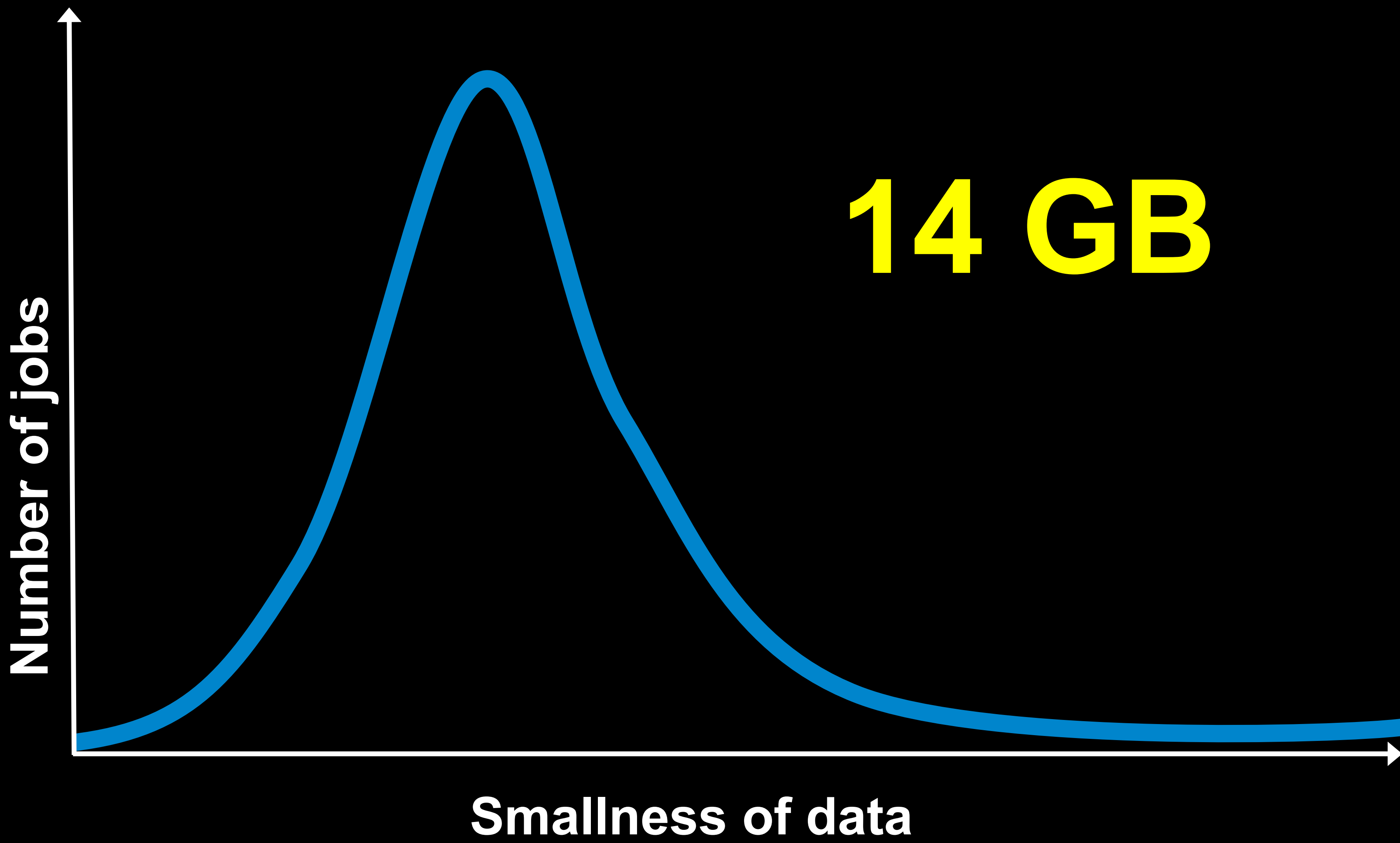
Analytics: This is really *raw data under storage*



Microsoft study of 174,000 analytic jobs in their cluster: median size ???

Third Nature

Working data for analytics most often not big



Third Nature

What makes data "big"?

Very large amounts

Hierarchical structures

Nested structures

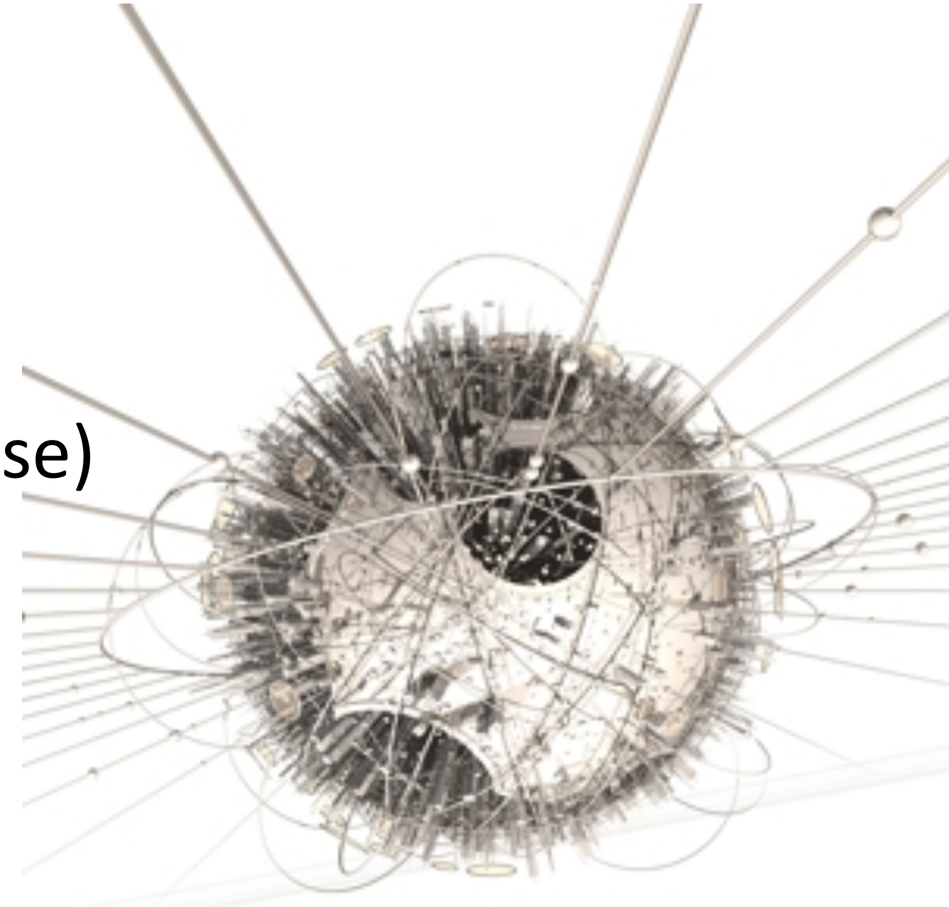
Encoded values

Non-standard (for a database) types

Time series

Deep structure

Human authored text



"big" is better off being defined as "complex" or "hard to manage"

Third Nature

Web tracking data has a nested structure

USER_ID	301212631165031
SESSION_ID	590387153892659
VISIT_DATE	1/10/2010 0:00
SESSION_START_DATE	1:41:44 AM
PAGE_VIEW_DATE	1/10/2010 9:59
DESTINATION_URL	https://www.phisherking.com/gifts/store/LogonForm?mmc=link-src-email-_-m100109-_-44IOJ1-_-shop&langId=-1&storeId=1055&URL=BECGiftListItemDisplay
REFERRAL_NAME	Direct
REFERRAL_URL	-
PAGE_ID	PROD_24259_CARD
REL_PRODUCTS	PROD_24654_CARD, PROD_3648_FLOWERS
SITE_LOCATION_NAME	VALENTINE'S DAY MICROSITE
SITE_LOCATION_ID	SHOP-BY-HOLIDAY VALENTINES DAY
IP_ADDRESS	67.189.110.179
BROWSER_OS_NAME	MOZILLA/4.0 (COMPATIBLE; MSIE 7.0; AOL 9.0; WINDOWS NT 5.1; TRIDENT/4.0; GTB6; .NET CLR 1.1.4322)

"unstructured" data embedded in the logged message: complex strings

…but it's easily unpacked into tuples

Third Nature

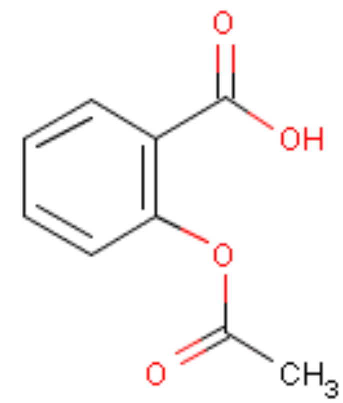
All of these things are “unstructured” data

Common Names

- Aspirin, Acetylsalicylic acid, Excedrin

Structural formulas

- Commonly used to communicate between chemists



Systematic nomenclatures:

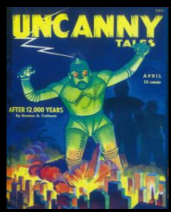
- Mass formula: C9H8O4
- SMILES: OC(=O)C1=C(C=CC=C1)OC(=O)C
- InChI: 1/C9H8O4/c1-6(10)13-8-5-3-2-4-7(8)9(11)12/h2-5H,1H3,(H,11,12)
- IUPAC: pyrido[1",2":1',2']imidazo[4',5':5,6]pyrazino[2,3-b]phenazine

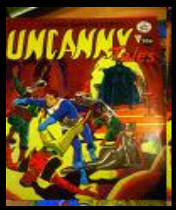
They all refer to the same thing.

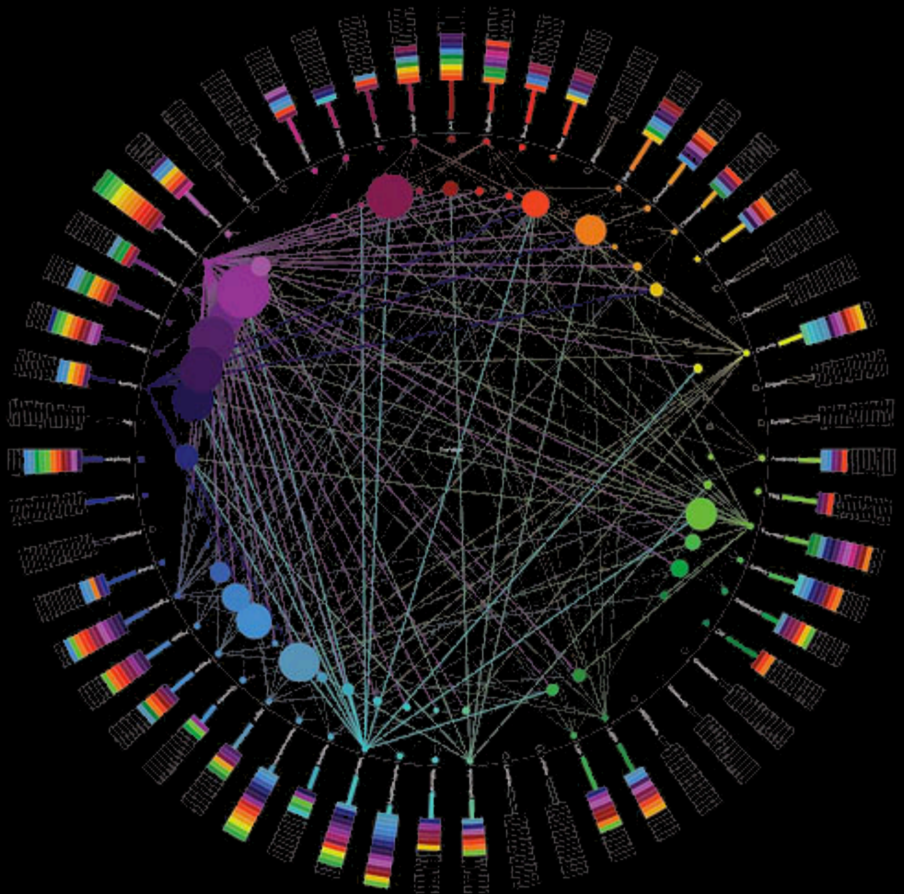
If you process them in a database, how do you store them?

Third Nature

We usually mean text, not data structures…







Unstructured data isn't really unstructured.

The problem is that this data is unmodeled.

The real challenge is complexity.

Third Nature

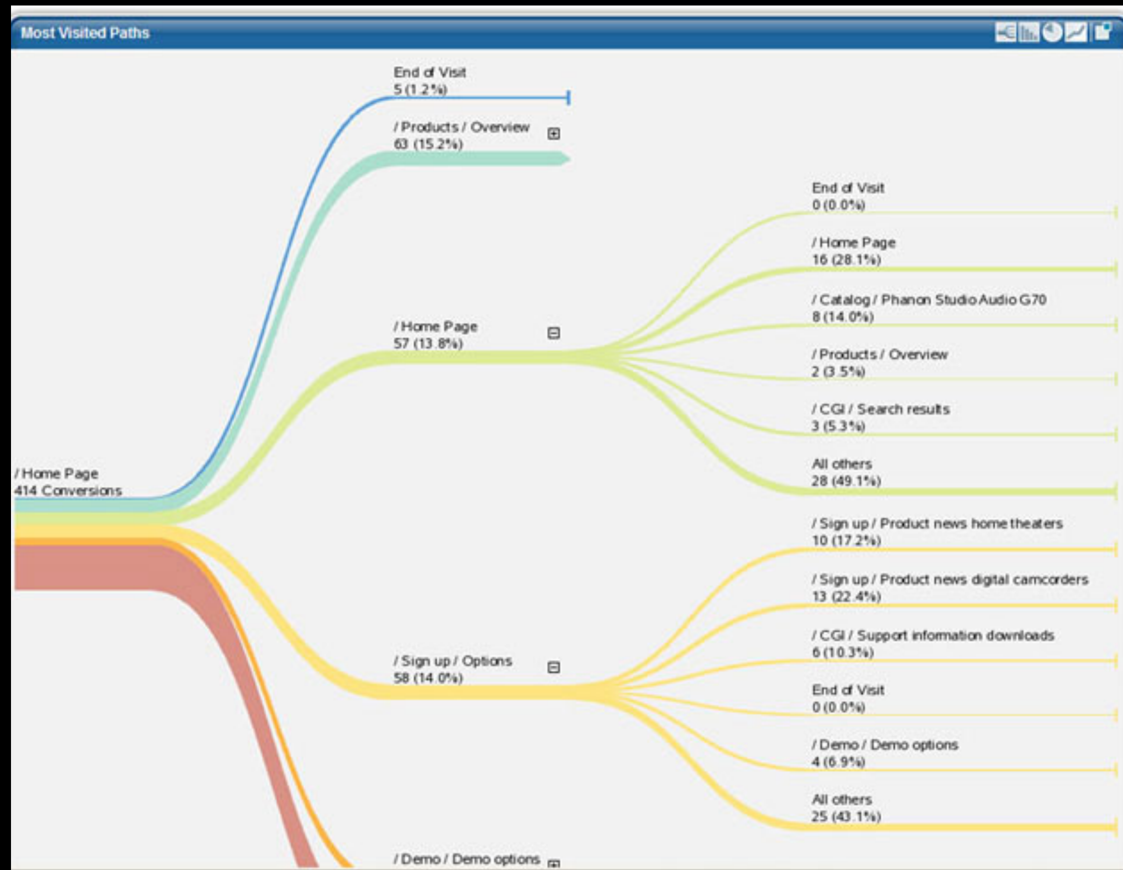
Patterns emerge from lots of event data

Patterns emerge from the underlying structure of the *entire dataset*.

The patterns are more interesting than sums and counts of the events.

Web paths: clicks in a session as network node traversal.

Email: traffic analysis producing a network



The event stream is a source for analysis, *generating another set of data* that is the source for different analysis.

Third Nature

BIGNESS AND DATA COMPUTATIONAL WORKLOADS

Not finished: remember the cycle of history…

The biggest hole in the prior sections is that we scaled OLTP and OLAP but what about analytics?

Queries <> transactions <> computations

The <u>three way</u> workload break

1. Operational: OLTP systems
2. Analytic: OLAP systems
3. Scientific: Computational systems

Unit of focus:

1. Transaction
2. Query
3. Computation

Different problems require different platforms

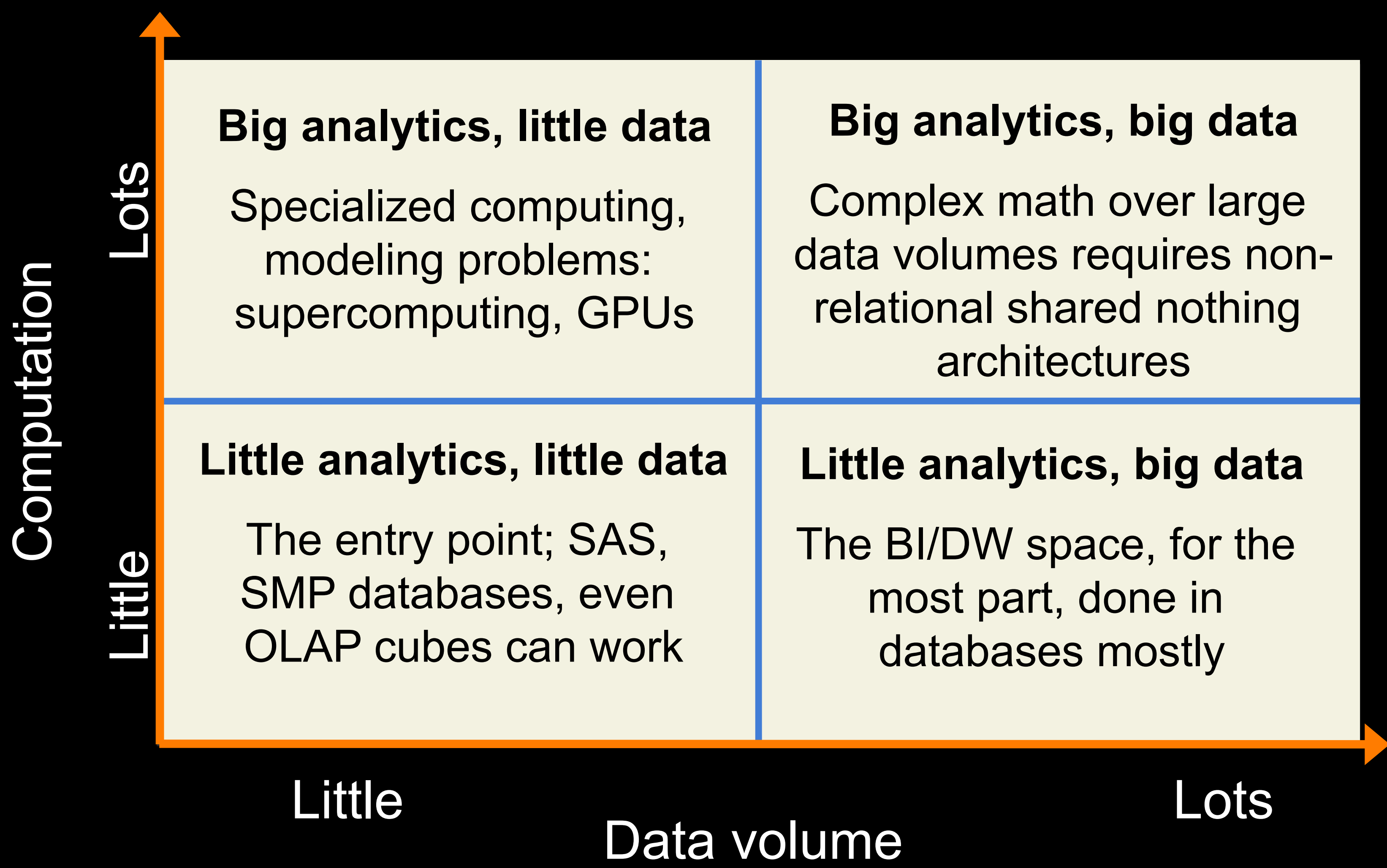
Third Nature

The holy grail of databases under current market hype

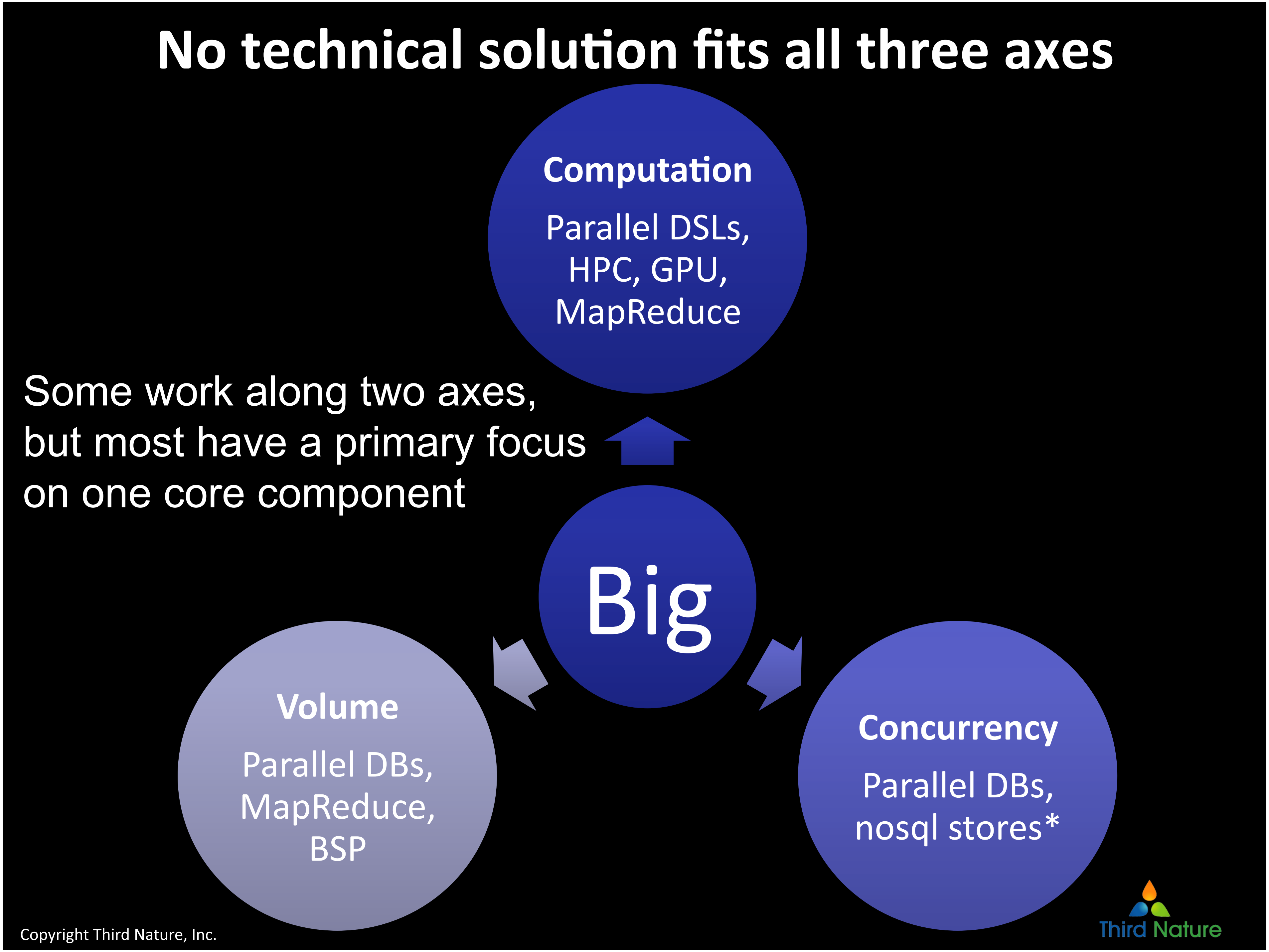
We're talking mostly about computation over data when we talk about "big data" and analytics.

The goal is combining data storage, retrieval and analysis into one system, *a potential mismatch for both relational and nosql.*

A Simple Division of the Analytic Problem Space

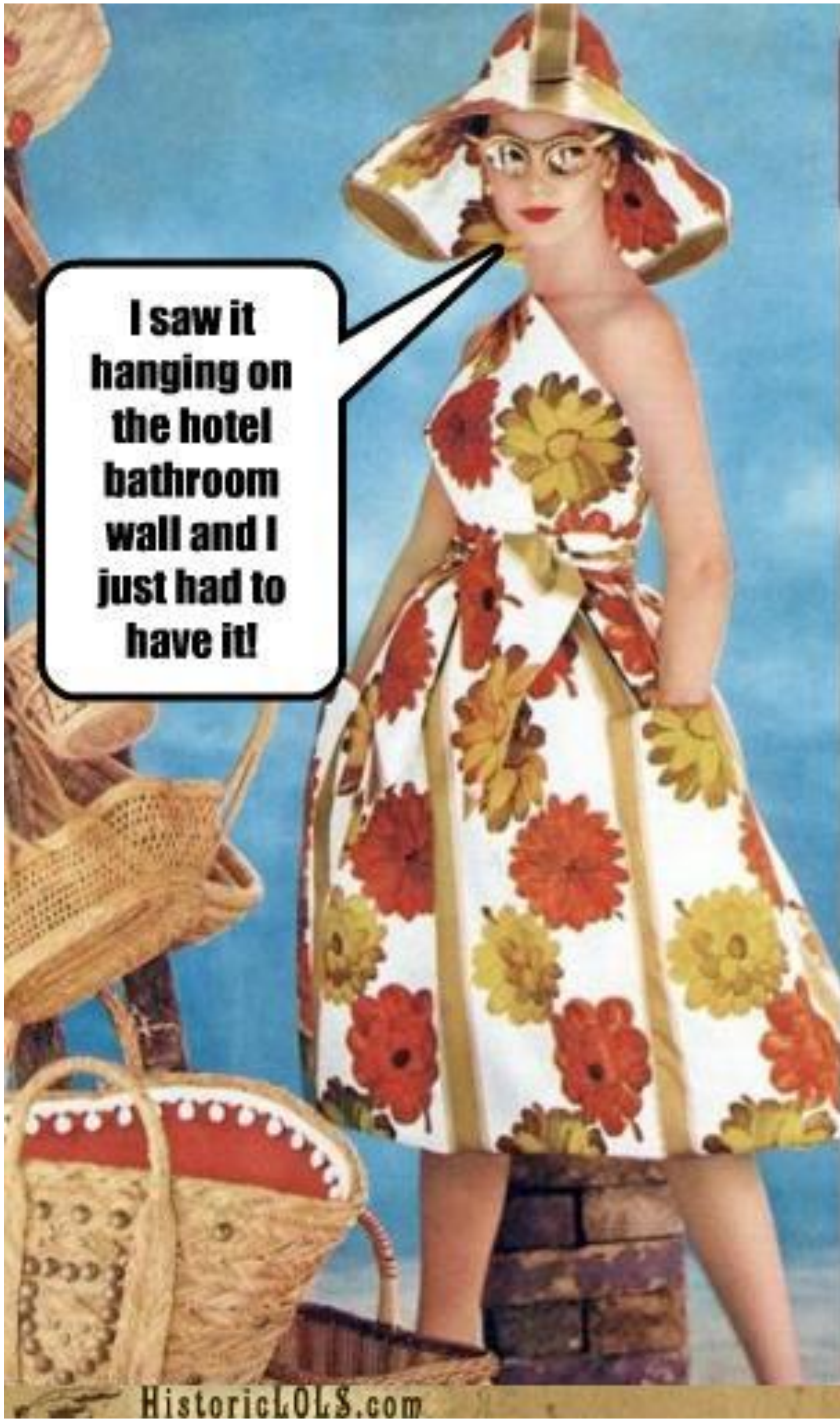


Third Nature


No technical solution fits all three axes
Computation
Parallel DSLs, HPC, GPU, MapReduce
Some work along two axes, but most have a primary focus on one core component
Big
Volume
Parallel DBs, MapReduce, BSP
Concurrency
Parallel DBs, nosql stores*
Copyright Third Nature, Inc.
Third Nature

Data infrastructure is a platform

- Any data – structures, forms
- Any latency –in motion, at rest
- Any process – query, algorithm, engine, transformation
- Any access – SQL, API, queue, file movement



Hadoop & NoSQL Adoption

Some people can't resist getting the next new thing because it's new.

Many organizations are like this, promoting a solution and hunting for the problem that matches it.

Better to ask "What is the problem for which this technology is the answer?"

Third Nature

NoSQL Will "Fail"

Unless some mathematically derived data model is developed, and a query language using it is created.

Otherwise there's no standard interfacing model, no interoperability, no chance for a tool ecosystem to evolve over top of the platform – because there are no uniform platform boundaries.

"One logical interface, many physical implementations" is a key reason why SQL won the database wars. This creates an ecosystem.

Maybe…


These aren't the databases we're looking for.
We need one that speaks pig latin.

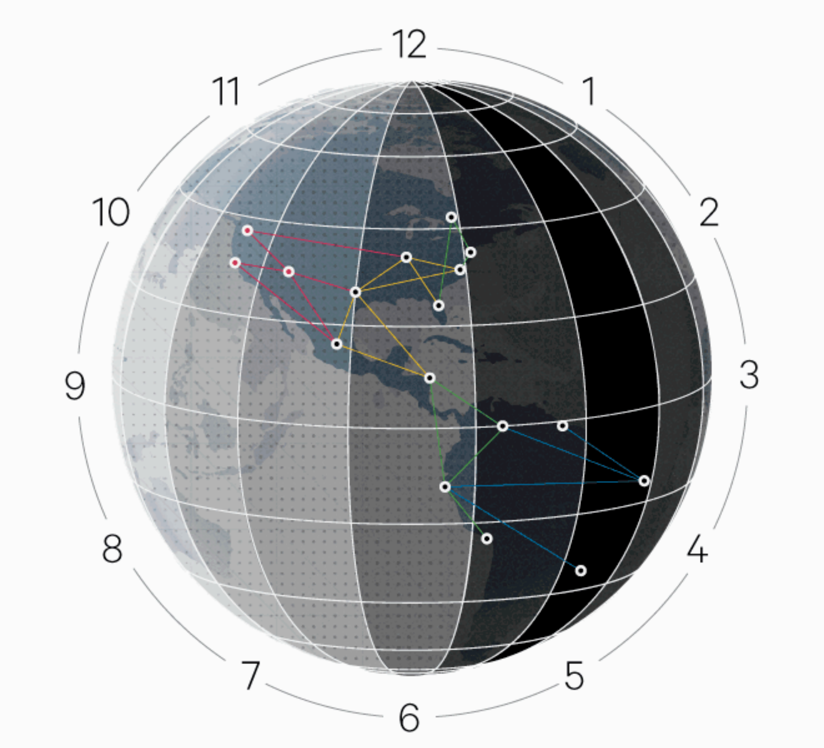
What's wrong with BASE?

Designing applications to cope with concurrency anomalies in their data is very error-prone, time-consuming, and ultimately not worth the performance gains.

developers spend a significant fraction of their time building extremely complex and error-prone mechanisms to cope with eventual consistency and handle data that may be out of date. We think this is an unacceptable burden to place on developers and that consistency problems should be solved at the database level. Full transactional consistency is one

Third Nature

Google F1: Another Evolution



Distributed SQL database

ACID compliance, 2PC and row-level locking (!)

Transparent data distribution

Synchronous replication across data centers

Table interleaving (hierarchies)

Queryable protobufs

MapReduce access to underlying data

Average user-facing latency of ~200ms with small deviation

Third Nature

The big data revolution, more of an evolution

Be pragmatic, not dogmatic

Conclusion


IF YOU
PROCRASTINATE
LONG ENOUGH
MOST PROBLEMS
SOLVE THEMSELVES
CRACKED.com

References (things worth reading on the way home)

A relational model for large shared data banks, Communications of the ACM, June, 1970, http://www.seas.upenn.edu/~zives/03f/cis550/codd.pdf

Column-Oriented Database Systems, Stavros Harizopoulos, Daniel Abadi, Peter Boncz, VLDB 2009 Tutorial http://cs-www.cs.yale.edu/homes/dna/talks/Column_Store_Tutorial_VLDB09.pdf

Nobody ever got fired for using Hadoop on a cluster, 1st International Workshop on Hot Topics in Cloud Data ProcessingApril 10, 2012, Bern, Switzerland.

A co-Relational Model of Data for Large Shared Data Banks, ACM Queue, 2012, http://queue.acm.org/detail.cfm?id=1961297

A query language for multidimensional arrays: design, implementation and optimization techniques, SIGMOD, 1996

Probabilistically Bounded Staleness for Practical Partial Quorums, Proceedings of the VLDB Endowment, Vol. 5, No. 8, http://vldb.org/pvldb/vol5/p776_peterbailis_vldb2012.pdf

"Amorphous Data-parallelism in Irregular Algorithms", Keshav Pingali et al

MapReduce: Simplified Data Processing on Large Clusters, http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en//archive/mapreduce-osdi04.pdf

Dremel: Interactive Analysis of Web-Scale Datasets, Proceedings of the VLDB Endowment, Vol. 3, No. 1, 2010 http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en//pubs/archive/36632.pdf

Spanner: Google's Globally-Distributed Database, SIGMOD, May, 2012, http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/es//archive/spanner-osdi2012.pdf

F1: A Distributed SQL Database That Scales, Proceedings of the VLDB Endowment, Vol. 6, No. 11, 2013, http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en/us/pubs/archive/41344.pdf

Third Nature

CC Image Attributions

Thanks to the people who supplied the creative commons licensed images used in this presentation:

shady_puppy_sales.jpg - http://www.flickr.com/photos/brizzlebornandbred/5001120150
cuneiform_proto_3000bc.jpg - http://www.flickr.com/photos/takomabibelot/3124619443/
cuneiform_undo.jpg - http://www.flickr.com/photos/charlestilford/2552654321/
scroll_kerouac.jpg - http://www.flickr.com/photos/ari/93966538/
House on fire - http://flickr.com/photos/oldonliner/1485881035/
Manuscripts on shelf - http://flickr.com/photos/peterkaminski/1688635/
manuscript_illum.jpg - http://www.flickr.com/photos/diorama_sky/2975796332/
manuscript_page.jpg - http://www.flickr.com/photos/calliope/306564541/
subway dc metro - http://flickr.com/photos/musaeum/509899161/

About the Presenter

Mark Madsen is president of Third Nature, a research and consulting firm focused on building the infrastructure for analytics, evidence-based management, business intelligence and data management. Mark is an award-winning author, architect and CTO whose work has been featured in numerous industry publications. Over the past ten years Mark received awards for his work from the American Productivity & Quality Center, TDWI, and the Smithsonian Institute. He is an international speaker, a contributor at Forbes Online and Information Management. For more information or to contact Mark, follow @markmadsen on Twitter or visit http://ThirdNature.net

Third Nature

About Third Nature

Third Nature is a research and consulting firm focused on new and emerging technology and practices in business intelligence, analytics and performance management. If your question is related to BI, analytics, information strategy and data then you‘re at the right place.

Our goal is to help companies take advantage of information-driven management practices and applications. We offer education, consulting and research services to support business and IT organizations as well as technology vendors.

We fill the gap between what the industry analyst firms cover and what IT needs. We specialize in product and technology analysis, so we look at emerging technologies and markets, evaluating technology and hw it is applied rather than vendor market positions.