

# TO SQL OR NOSQL?

---

That is the question

DAVID OSTROVSKY  
Proofpoint

# Newer is always better, right?



SQL

Relational  
Known  
Mature



NoSQL

Flexible  
New  
Sexy

# Taxonomy

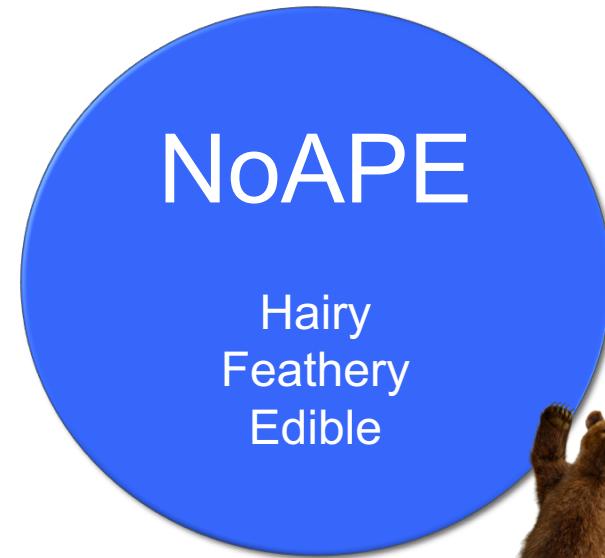
APE

Hairy  
Smart  
Lovable



NoAPE

Hairy  
Feathery  
Edible



# Documents and objects and graphs, oh my!

## Key-Value



Memcached



redis



Couchbase



mongoDB



Google  
BigTable

AEROSPIKE



RavenDB



Microsoft  
SQL Server



Neo4j



GraphBase



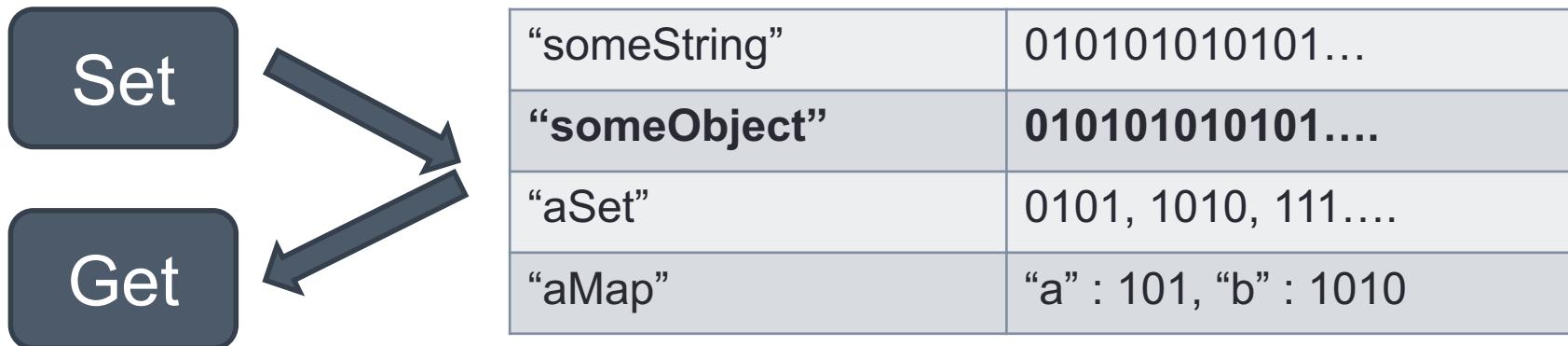
InfiniteGraph



TITAN

# Key-Value Stores

```
var someString = "Hello, I'm a string!";  
var someObject = new User { ... };  
var aSet = new Set { "a", "b", "c" };
```



# Documents and objects and graphs, oh my!

## Key-Value



Memcached



redis



## Document



Couchbase



mongoDB



Amazon  
DynamoDB



RavenDB



Google  
BigTable

AEROSPIKE



Microsoft  
SQL Server



Neo4j



GraphBase



InfiniteGraph

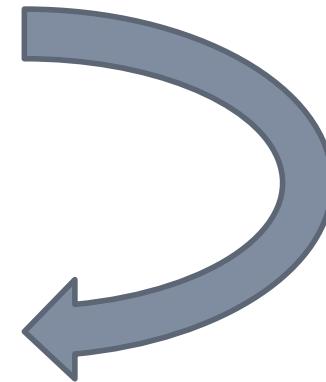


TITAN

# Document Stores

UserId	Name	Age	CityId
01	John	35	002
02	Dave	18	003

```
{  
    "name": "John",  
    "age": 35,  
    "address": {  
        "city": "New York",  
        "street": "..."  
    }  
}
```



# Documents and objects and graphs, oh my!

## Key-Value



Memcached



redis



## Document



Couchbase



mongoDB



Amazon  
DynamoDB



RavenDB

## Row



Google  
BigTable



Microsoft  
SQL Server



Neo4j



GraphBase



InfiniteGraph



TITAN

AEROSPIKE

# Partitioned Row Stores

Row Key	Personal			Address	
User Id	Name	Age	City	Zip	
01	John	41	New York	10001	
02	Dave	36	Seattle	98101	
			Version 1	34	
			Version 2	35	
			Version 3	36	

# Documents and objects and graphs, oh my!

## Key-Value



Memcached



redis



## Document



Couchbase



mongoDB



RavenDB

AEROSPIKE

## Row



Google BigTable

## Column



Microsoft SQL Server



Neo4j



GraphBase



InfiniteGraph



TITAN

# Column Oriented Databases

UserId	Name	Age	City
01	John	35	New York
02	Dave	18	Seattle
03	Mary	29	New York
04	Bob	47	London

... UserId:01 ; Name: John ; Age: 35 ; City: New York ...



... New York:001,003 ; Seattle:002 ; London: 004 ...

# Documents and objects and graphs, oh my!

## Key-Value



Memcached



redis



## Document



Couchbase



mongoDB



RavenDB

AEROSPIKE

## Row



Google BigTable

## Column



Microsoft SQL Server

## Graph



Neo4j



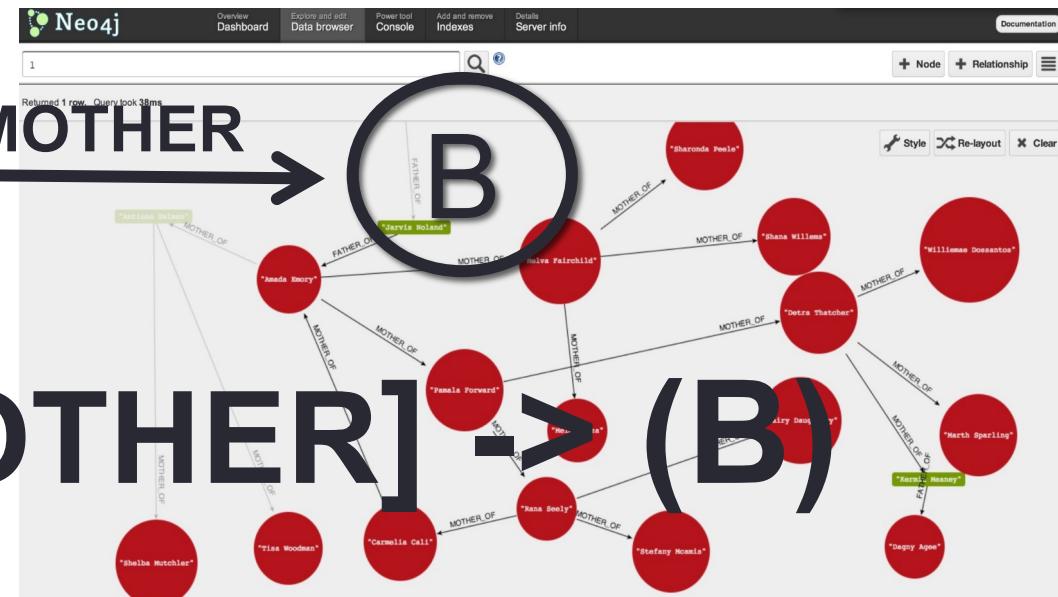
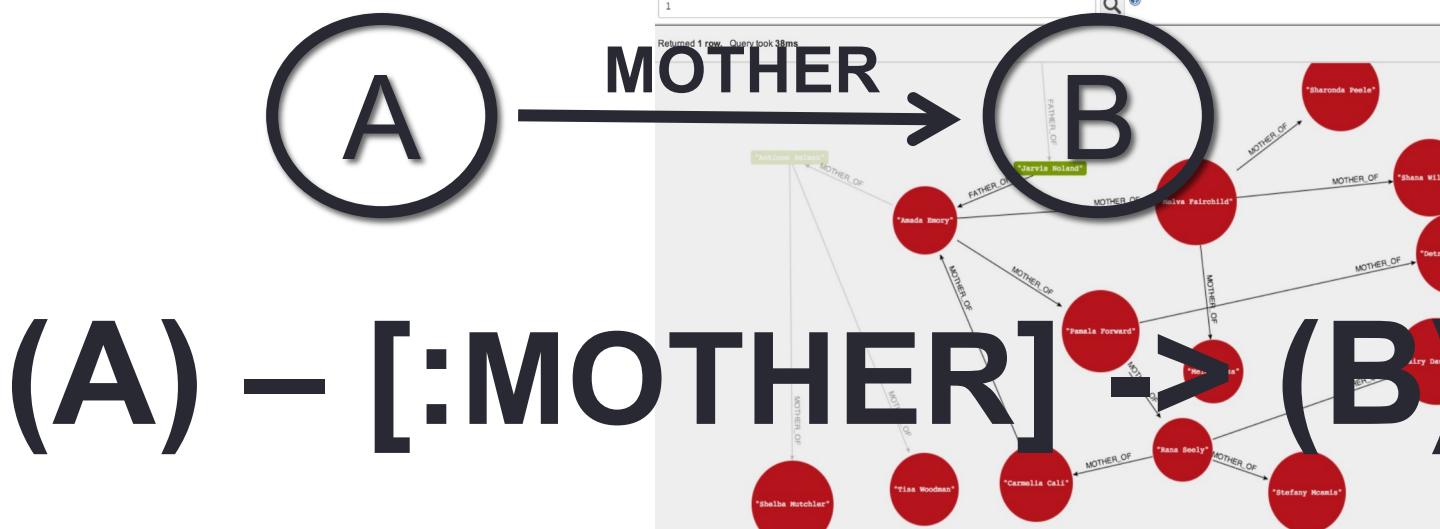
GraphBase



TITAN

# Graph Databases

Name	Type	From	To	Relationship Type
Amada Emory	Female	1	11	MOTHER_OF
Rana Seely	Female	1	12	MOTHER_OF



# NoSQL-ing the SQL

???



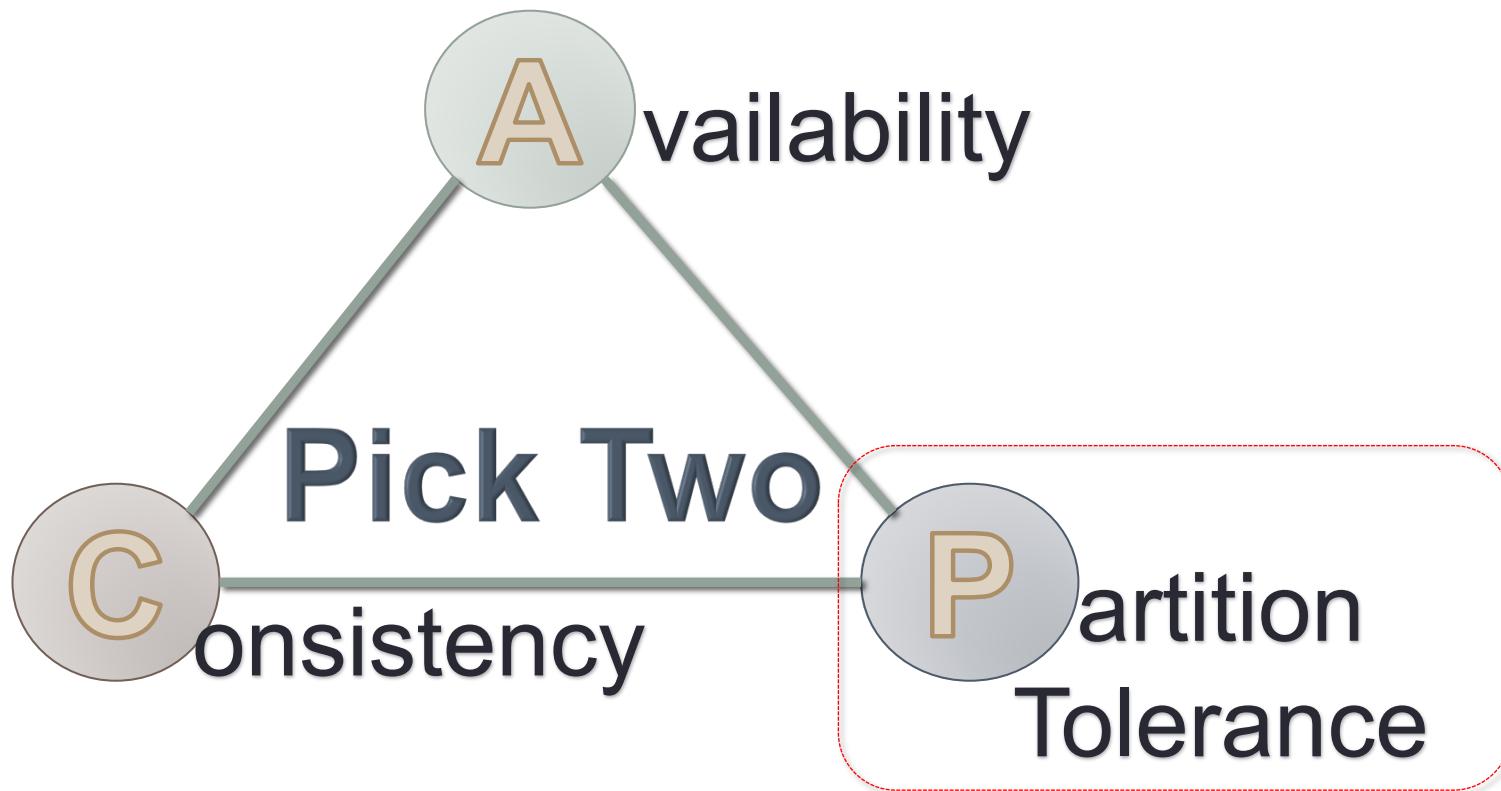
Cluster CGE



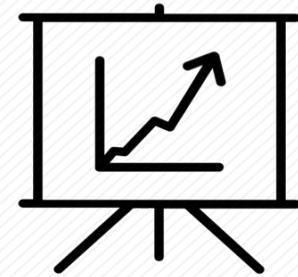
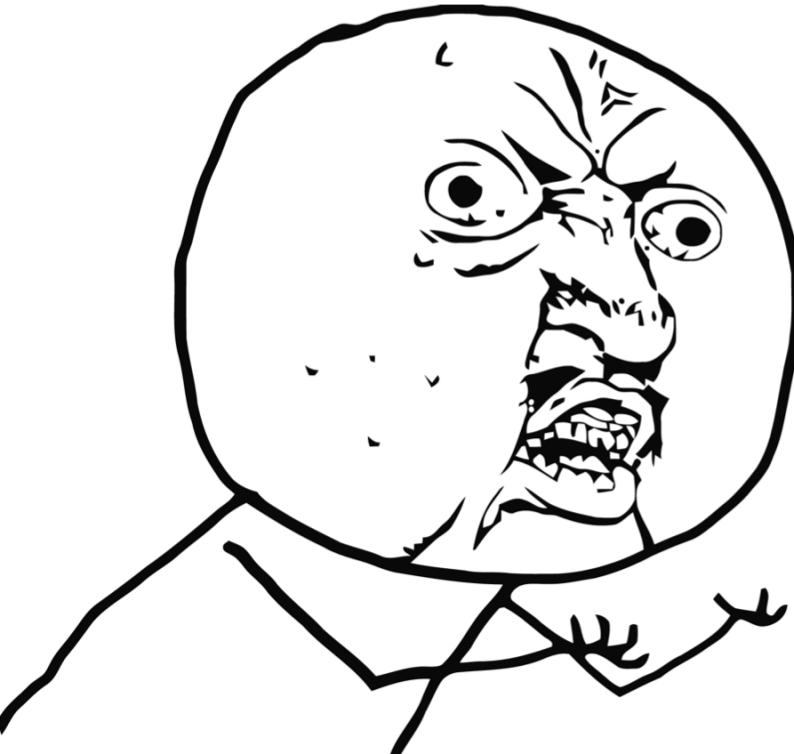
PostgreSQL

**ORACLE®**  
D A T A B A S E

# The CAP Theorem



# Benchmarks



Y U NO HELP?

# Search, a different beast altogether

SELECT \* FROM pictures WHERE animal = "Cat";



About 261,000,000 results (0.39 seconds)

Showing results for find me pictures of **cats**

Search instead for find me pictures of kats

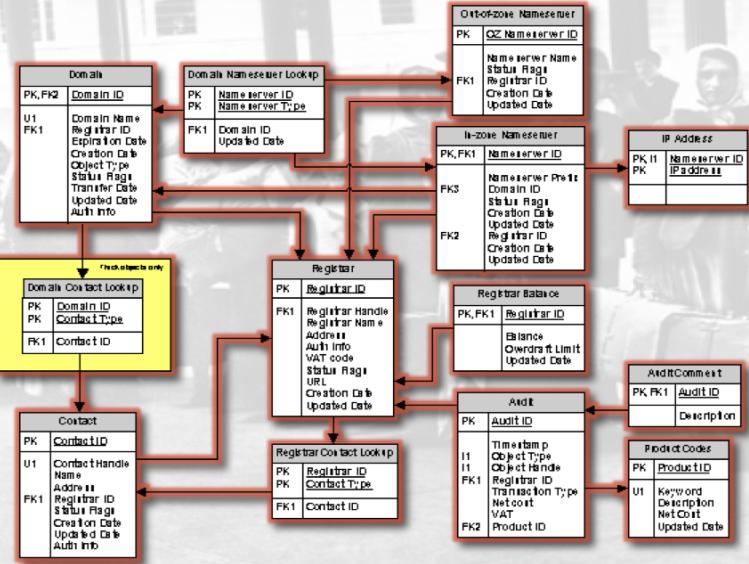
---

[Images for cats](#)

[Report images](#)



# Migrations are Hard



## Document

Lorem ipsum dolor sit amet.  
Consectetur adipiscing elit. Cras non nunc nec enim tristique blandit. Vestibulum quis tellus.  
Duis nulla. Donec iustus urna. Sed tempus nibh id massa. Vivamus placerat justo quis nibh. Ut quis ante. Ut sollicitudin quam eu mi. Donec  
incasse purus sit amet velit. Sed ac sem.  
Aenean quis justo. Vestibulum ante ipsum primis  
in faubibus erit luctus et ultrices posuere cubilia  
Curas; Ut blandit.  
Nulla facilisi. Aenean eros felis, blandit eu,  
commodo sit amet, varius a, pede. Curabitur  
augue felis, congue sed.

# Summing it all up

- New != Better
- Performance is in the eye of the beholder
- You are not Facebook\*
- One database to rule them all works for hobbits, not so much for giants



\* Unless you are.