



NoSQL



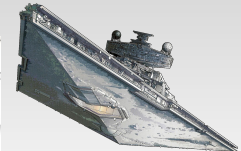
... la peor definición de la historia.



1998: NoSQL

RDBMS

Pero sin interfaz SQL



NOSQL meetup

Last.fm

Thursday, June 11, 2009 from 10:00 AM to 5:00 PM (PDT)

San Francisco, CA

Ticket Information

TYPE	REMAINING	END	QUANTITY
Free ticket	Sold Out	Ended	Free
			Sold Out

Share NOSQL meetup

Share Tweet Like One person likes this. Be the first of your friends.

Event Details

Introduction

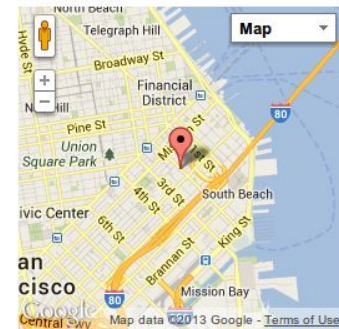
This meetup is about "open source, distributed, non relational databases".

Have you run into limitations with traditional relational databases? Don't mind trading a query language for scalability? Or perhaps you just like shiny new things to try out? Either way this meetup is for you.

Join us in figuring out why these newfangled Dynamo clones and BigTables have become so popular lately. We have gathered presenters from the most interesting projects around to give us all an introduction to the field.

Preliminary schedule

When & Where



CBS Interactive, Magma room
235 Second Street
San Francisco, CA 94105

Thursday, June 11, 2009 from 10:00 AM to 5:00 PM (PDT)



NoSQL no es una
tecnología, es un
movimiento



NoSQL: Bases distribuidas y no relacionales

NoSQL

Clave-Valor

key	value
-----	-------

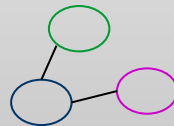
Columnares

key	att	att	
	value	value	
key	att	att	att
	value	value	value

Documentales

```
{  
  "id": "value",  
  "attribute": "value"  
}
```

Grafos





¿Por qué y cómo surgen?



Google



BigTable

amazon



Dynamo



- Escala web
 - Web 2.0
 - Generación de contenidos por parte del usuario
 - Aumento exponencial del volumen de datos
- Los datos evolucionan
- Hardware y sistemas pasan a ser un Commodity
- Post boom de las .com las empresas fueron más conservadoras en sus gastos



Problemas con los RDBMS

- La tecnología base tiene 40 años
- Menor cantidad de problemas a resolver
- Mainframe/Servidor
- Las relaciones (tablas) a veces no son el mejor modelo para nuestros datos
- Poca flexibilidad para introducir cambios
- Complejos cuándo necesitan **escalar**



Poca flexibilidad para
introducir cambios



ALTER TABLE ADD COLUMN takes a long time

Asked 10 years, 11 months ago Modified 5 months ago Viewed 83k times



88



I was just trying to add a column called "location" to a table (main_table) in a database. The command I run was

```
ALTER TABLE main_table ADD COLUMN location varchar (256);
```



41



The main_table contains > 2,000,000 rows. It keeps running for more than 2 hours and still not completed.

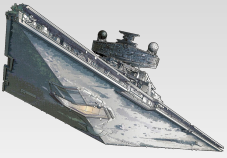


Agregar/eliminar columna/índice

- Los RDBMS tienen un límite de columnas
 - SQL Server ~ 1024
 - Postgres ~ 1600
 - MySQL ~ 4096
 - Oracle ~ 1000
 - DB2 ~ 8000



Dispersión

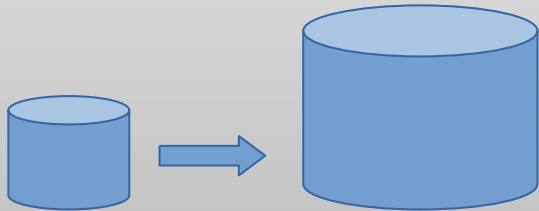


Normalizar los atributos



Escalar: Adaptarse al aumento de demanda sin perder calidad

Vertical



Horizontal





¿Cómo escalar un RDBMS?

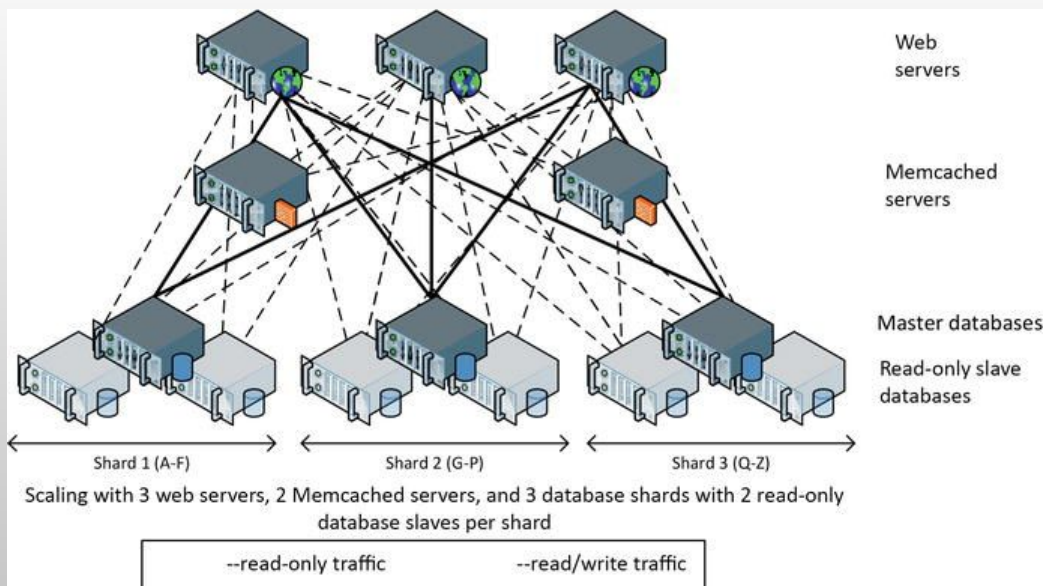
- Mejorar el hardware
- Eventualmente caro
- Replicación Master-Slave
 - Complejo
 - Introduce un single point of failure
- Sharding
 - Más complejo todavía



Si el volumen de datos
aumenta, divide y
vencerás



Sharding



Facebook: Más de 4.000 shards en 2011²⁰



Las consultas SQL
funcionan en un único
shard



Es complejo
mantener la
consistencia en
varios servidores

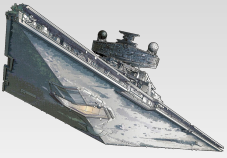


Teorema CAP

Consistency

Availability

Partition Tolerance



En un sistema distribuido,
un nodo siempre va a
fallar (P)

Consistencia (C) o Disponibilidad(A)



Escalaron con clusters de hardware barato y sistemas distribuidos



¿Y qué pasa con la
consistencia?



ACID vs BASE



BASE

Basically

Available

Soft state

Eventually consistent



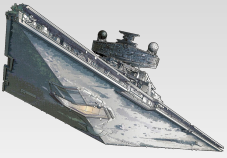
BASE: El sistema está disponible y en algún momento va a estar consistente



El negocio elige si es C o A



CAP aplica solo para
problemas ¿pero en la
operativa normal?



¿Consistencia o latencia?



¿Cómo elegimos?



Depende



- Muchos datos (realmente muchos)
- Performance en escritura, volumen masivo de datos
- Lectura rápida con clave-valor
- Esquemas flexibles
- Migración de esquemas
- Eliminar el único punto de fallo
- Paralelismo
- Facilidad para el desarrollador



El desafío es elegir el
modelo de datos correcto
para el dominio del problema

One solutions does not fit all