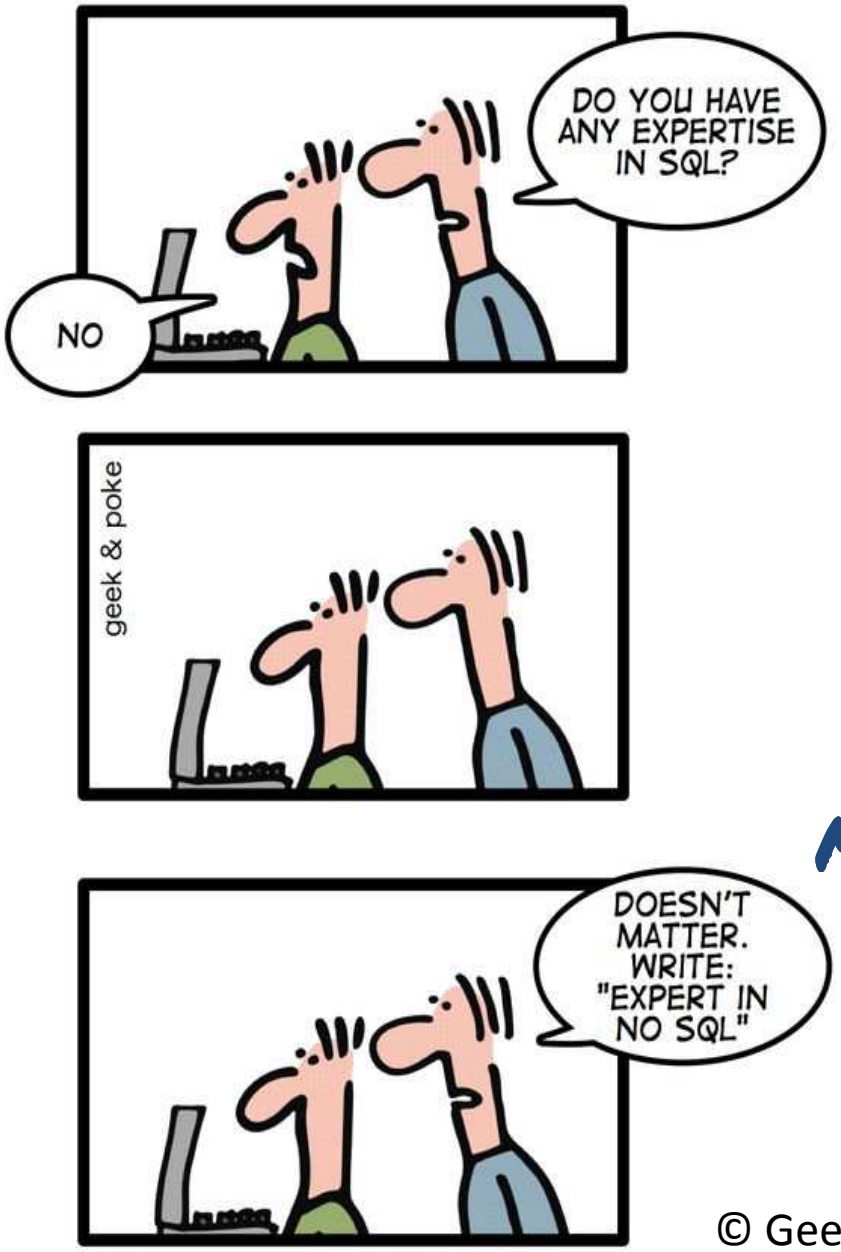
HOW TO WRITE A CV

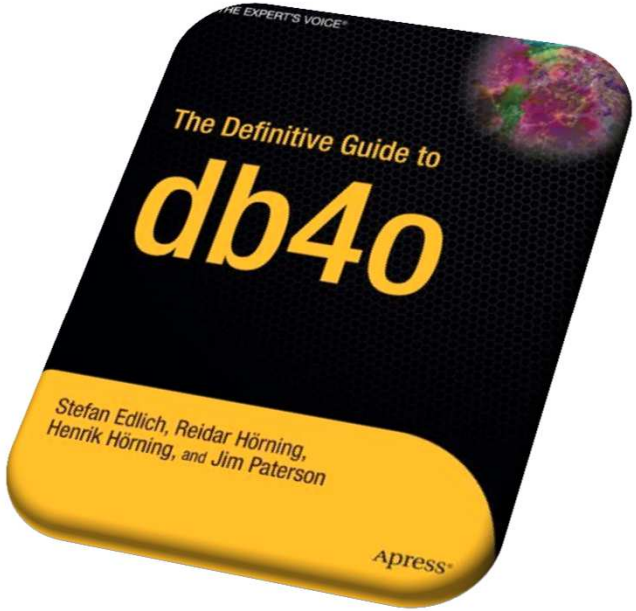


THE 10 WHYS OF NOSQL

NOSQL ROADSHOW

MARCH 2012 COPENHAGEN

© Geek&Poke



nosqlberlin.de
nosqlfrankfurt.de
nosql powerdays

N★SQL matters

HTTP://NOSQL-DATABASE.ORG

NoSQL

Your Ultimate Guide to the Non - Relational Universe!

[the **biggest** nosql link Archive in the web]
...never miss a conceptual article again...
News Feed covering all changes here !

NoSQL DEFINITION: Next Generation Databases mostly addressing some of the points: being **non-relational, distributed, open-source** and **horizontal scalable**. The original intention has been **modern web-scale databases**. The movement began early 2009 and is growing rapidly. Often more characteristics apply as: **schema-free, easy replication support, simple API, eventually consistent** / **BASE** (not ACID), a **huge data amount**, and more. So the misleading term *"nosql"* (the community now translates it mostly with "**not only sql**") should be seen as an alias to something like the definition above.
[based on 5 sources, 10 constructive feedback emails (thanks!) and1 insulting comment. Agree / Disagree? Tell me so!]

LIST OF NOSQL DATABASES [beta]

Core NoSQL Systems:

Wide Column Store / Column Families

Hadoop / HBase: API: **Java / any writer**, Protocol: **any write call**, Query Method: **MapReduce Java / any exec**, Replication: **HDFS Replication**, Written in: **Java**, Concurrency: ?, Misc: **Links**: 3 Books [1, 2, 3]

Cassandra: API: **many Thrift » languages**, Protocol: ?, Query Method: **MapReduce**, Replicaton: , Written in: **Java**, Concurrency: **eventually consistent** , Misc: like "Big-Table on Amazon Dynamo alike", initiated by Facebook, Slides » , Clients », Installation »

Hypertable: API: **Thrift** (Java, PHP, Perl, Python, Ruby, etc.), Protocol: **Thrift**, Query Method: **HQL, native Thrift API**, Replication: **HDFS Replication**, Concurrency: **MVCC**, Consistency Model: **Fully consistent** Misc: High performance C++ implementation of Google's Bigtable. Commercial support »

Cloudera: Professional Software & Services based on Hadoop.

Amazon SimpleDB: **Misc**: not open source / part of AWS, Book »

[SciDB: **Array** Data Model for Scientists, paper », poster », HiScaBlog »]
[OpenNeptune, Qbase, KDI]:

EVENTS

17th JAN **Membase** Seattle »
5th JAN **Membase** San Diego
3rd DEC **Mongo** Mountain View
20th NOV NoSQL **Dundee** »
All past NoSQL Conferences »
register your event here! »

NoSQL Summer in 27 cities »

Conceptual quality articles, links, research papers, etc.

NoSQL

ARCHIVE

Announcing the availability of the **worlds first NoSQL Book**!

NoSQL FORUMS [3]
* Global NOSQL Forum »
* Forum Berlin »
* Forum France »
* Forum Japan »

GREAT NoSQL NEWS FEEDS
* MyNoSQL by Alex P »
[He is definetely bigger...]
* On Twitter: nosqlupdate »
* HighScalability Blog »

BREAKING NoSQL NEWS

2011

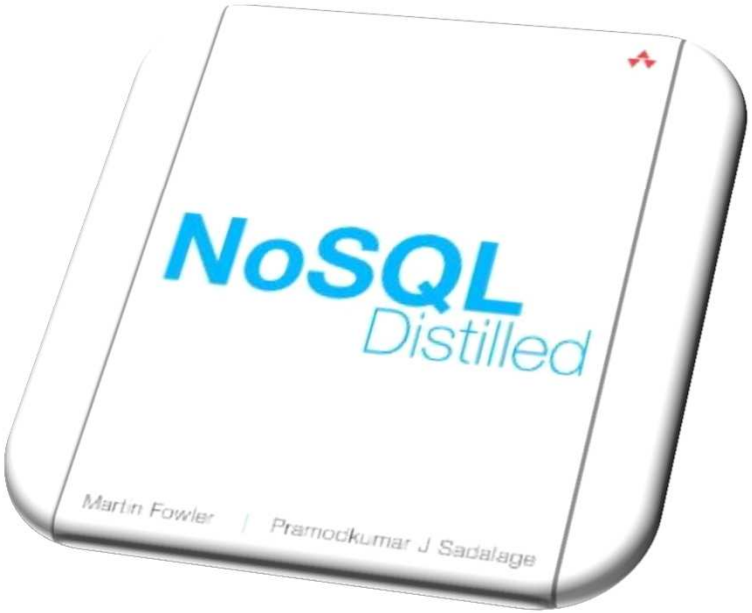
The NoSQL Year!

Developer Skills

1. HTML5
2. MongoDB
3. iOS
4. Android
5. Mobile app
6. Puppet
7. Hadoop
8. jQuery
9. PaaS
10. Social Media

Professional
NoSQL
Shashank Tiwari

Seven Databases
in Seven Weeks
A Guide to Modern Databases
and the NoSQL Movement
Eric Redmond
and Jim R. Wilson
BET


NoSQL
Distilled
Martin Fowler
Pramodkumar J Sadalage

Redis Cluster soon

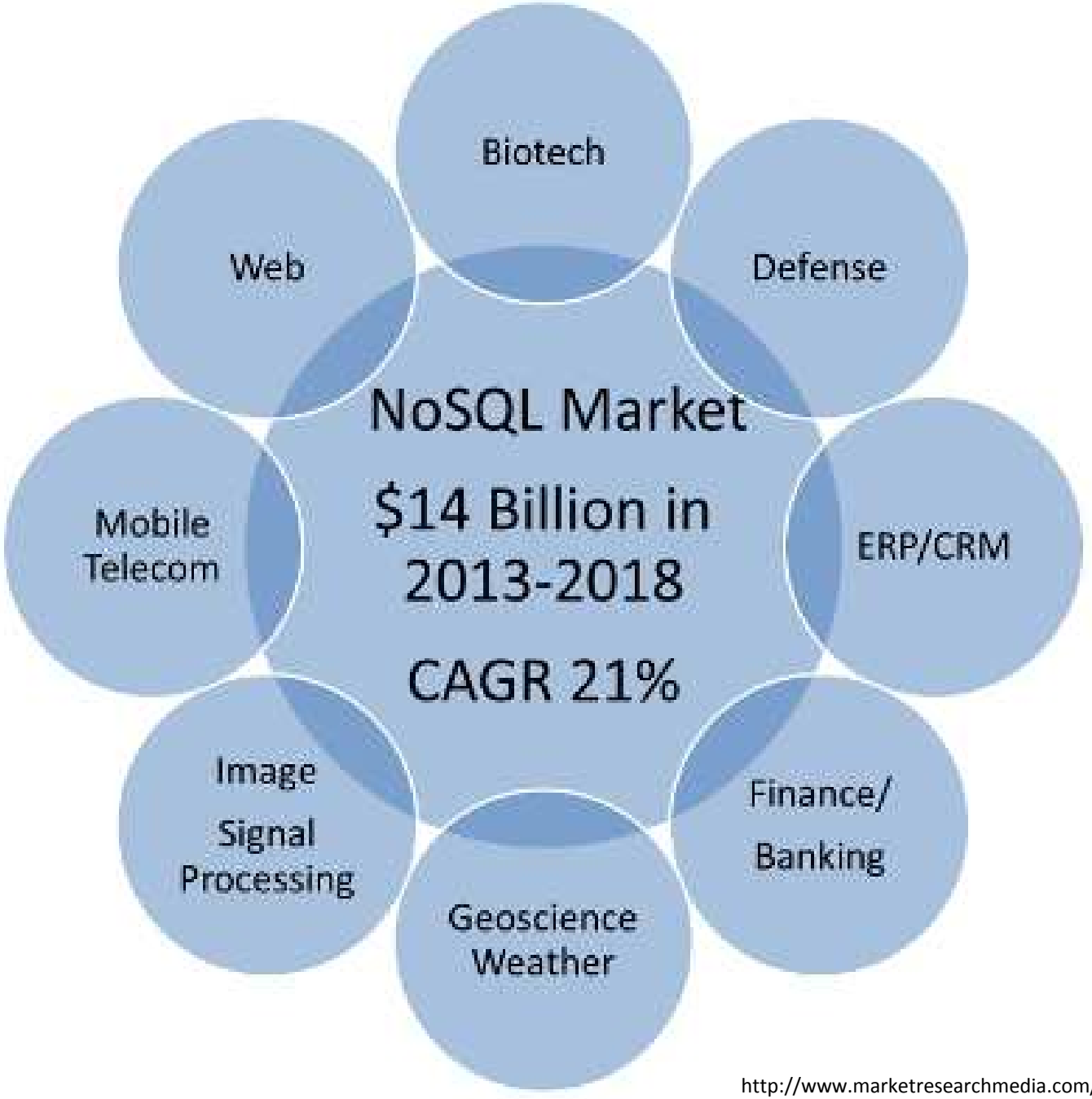
Hadoop + Cassandra 1.0

MongoDB 2.0 + MMS

Riak 1.1

>1000 hosting Services

NoSQL is PaaS Standard!



http://www.marketresearchmedia.com/2010/11/11/nosql-market/

Oracle's Thomas Kurian says no to SQL. Sorta. (Photo: Oracle/Flickr)

ORACLE®

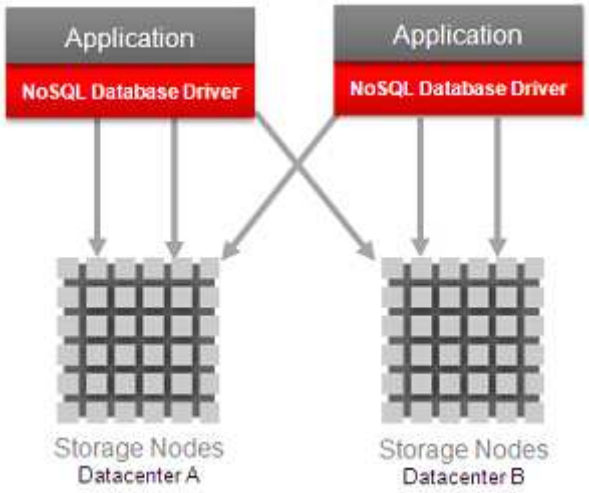
summer 2011

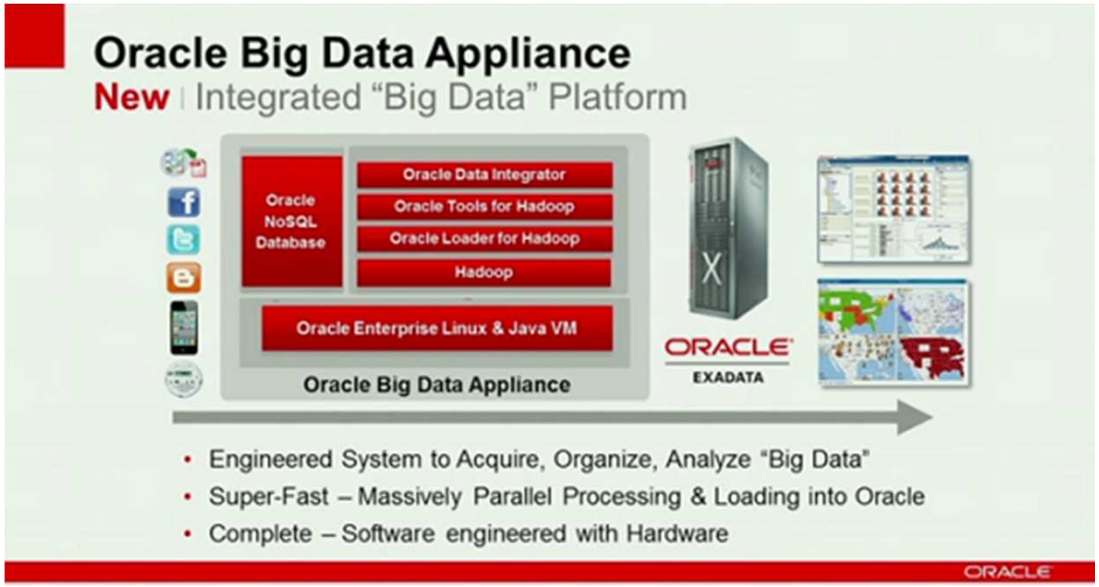
white paper:
„debunking the (NoSQL) hype“

“The NoSQL databases are beginning to feel like an ice cream store that entices you with a new flavor of the month,” the white paper read. “[But] you shouldn’t get too attached to any of the flavors because it may not be around for too long.”

ORACLE®

Oracle NoSQL Database

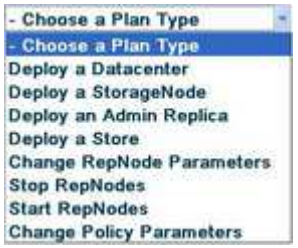




ConsHash
config ACID
no single PF
DataC Replication
Top Admin
"BerkleyDB reloaded"

cloudera

Hadoop + Manager



Microsoft

Windows Azure

Microsoft
SQL Server

+=
hadoop

user defined functions in C++ & Java
⇨10x faster then SQL or Stored Procs

UDF connector for Hadoop ⇨ ☺

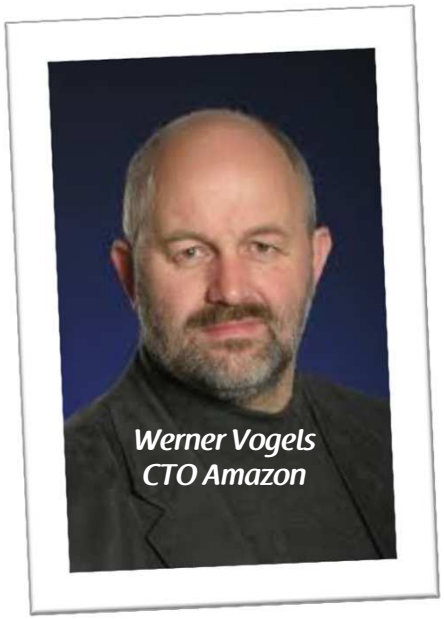
C++ APIs for Map Reduce ⇨ ☺

Greenplum, Pervasive and 100 others too…

WHY ?

nosqltapes.com

NoSQL is specialization!



70% Scaling

30% Business

70% Business

30% Scaling

70% KV

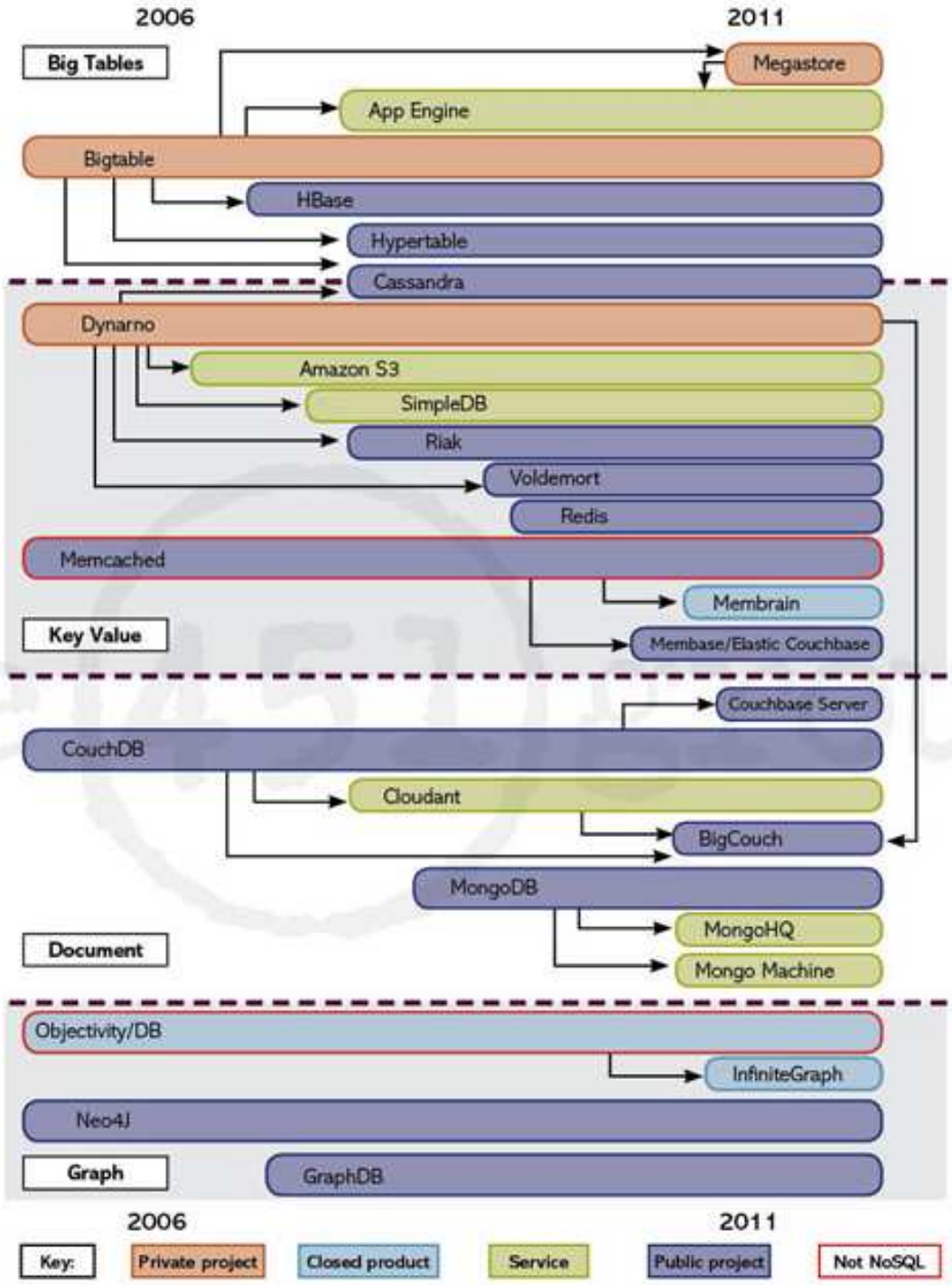
20% SingleTable

10% Joins

interaction vs. transaction is 1000 : 1

WHY?

Needs!


2006
2011
Big Tables
Megastore
App Engine
Bigtable
HBase
Hypertable
Cassandra
Dynarno
Amazon S3
SimpleDB
Riak
Voldemort
Redis
Memcached
Membrain
Membase/Elastic Couchbase
Key Value
Couchbase Server
CouchDB
Cloudant
BigCouch
MongoDB
MongoHQ
Mongo Machine
Document
Objectivity/DB
InfiniteGraph
Neo4J
Graph
GraphDB
2006
2011
Key:
Private project
Closed product
Service
Public project
Not NoSQL

80s / 90s

- Megabytes, Gigabytes
- Scale Out worked fine
- Golden Age for Oracle, DB2, MS, MySQL…

WHY #1 Big Data!

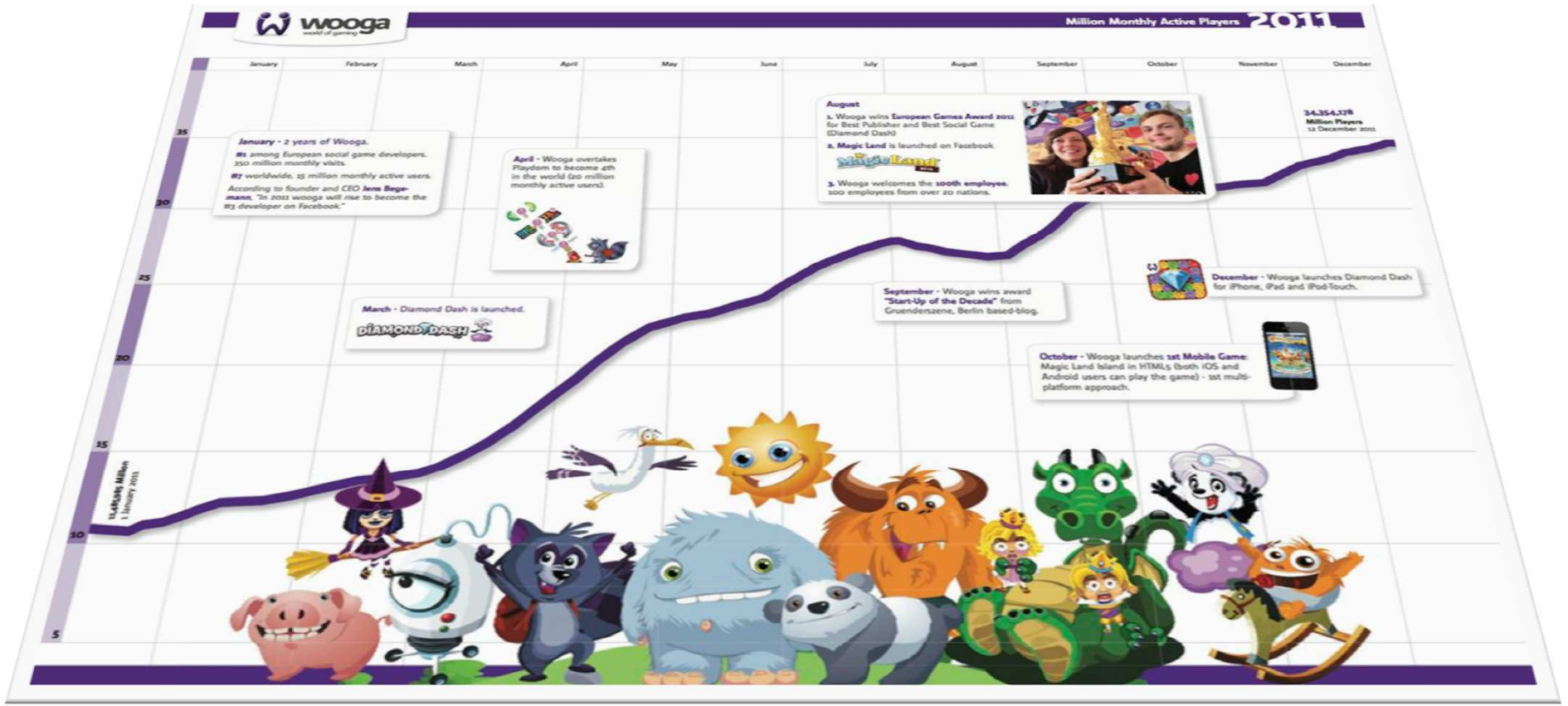
Terrabyte-, Petabyte-Age

Simple Scale-Out needed

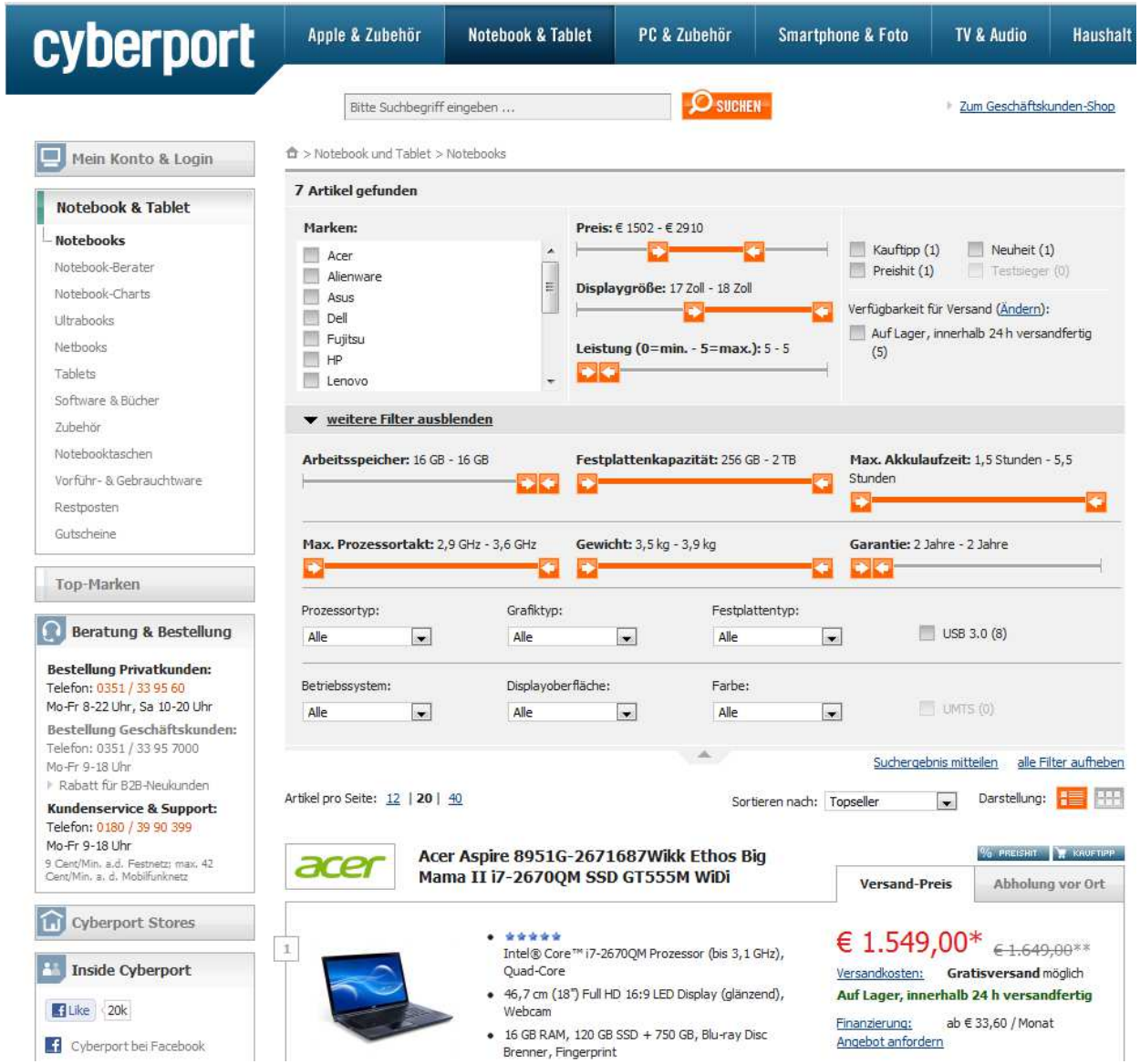
I am not Amazon?!

WHY #2

Massive Write Performance!
Write Availability!

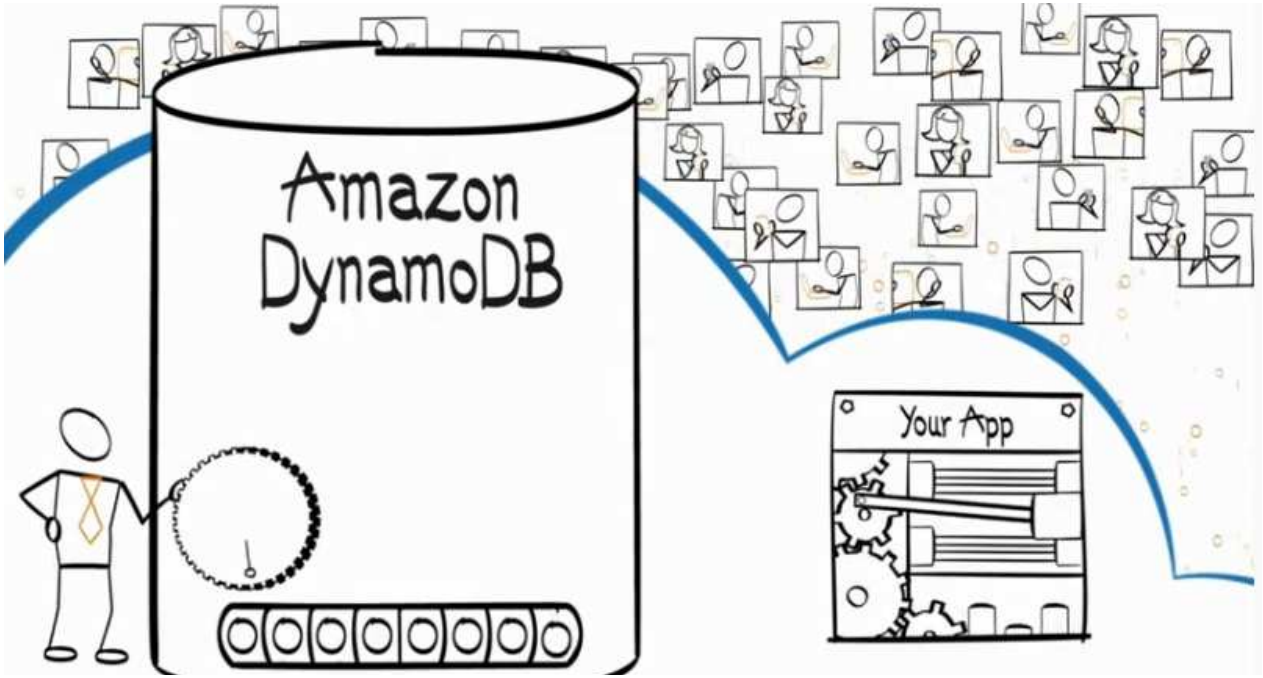


WHY #3 Sub Second Responses!



WHY #4 Fast KV Access

amazon.com



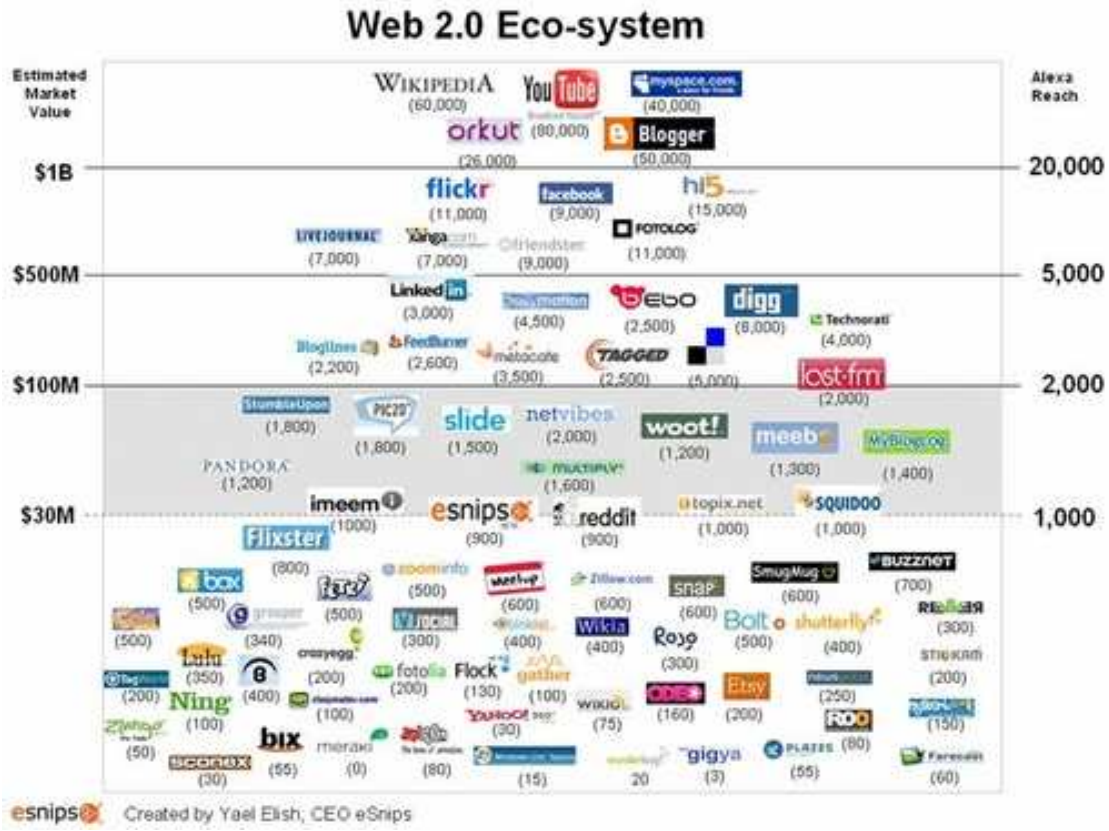
>100 Mio PVs

YOUPORN

WHY #5

Flexible Schema (Migration)

Flexible Datatypes

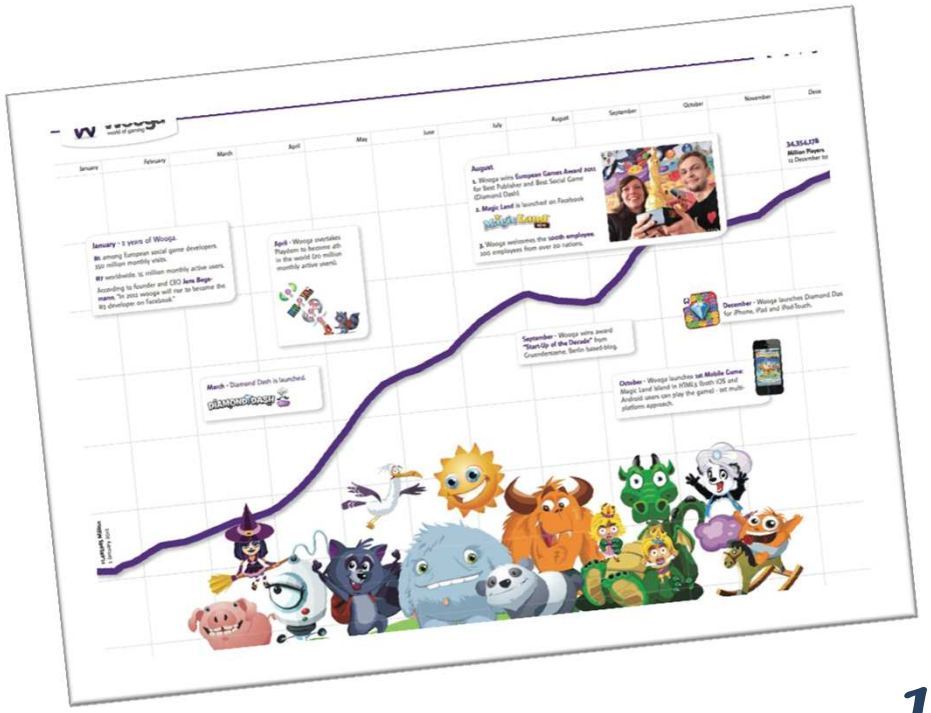


esnips Created by Yael Elish, CEO eSnips

ALTER TABLE => Shut Down for 3 hours in "the night?! "

WHY #6 easier -maintainability

-administration and operations



1 Game =

2 REDIS instances

+ 2 Replicas

WHY #7 no single point of failure

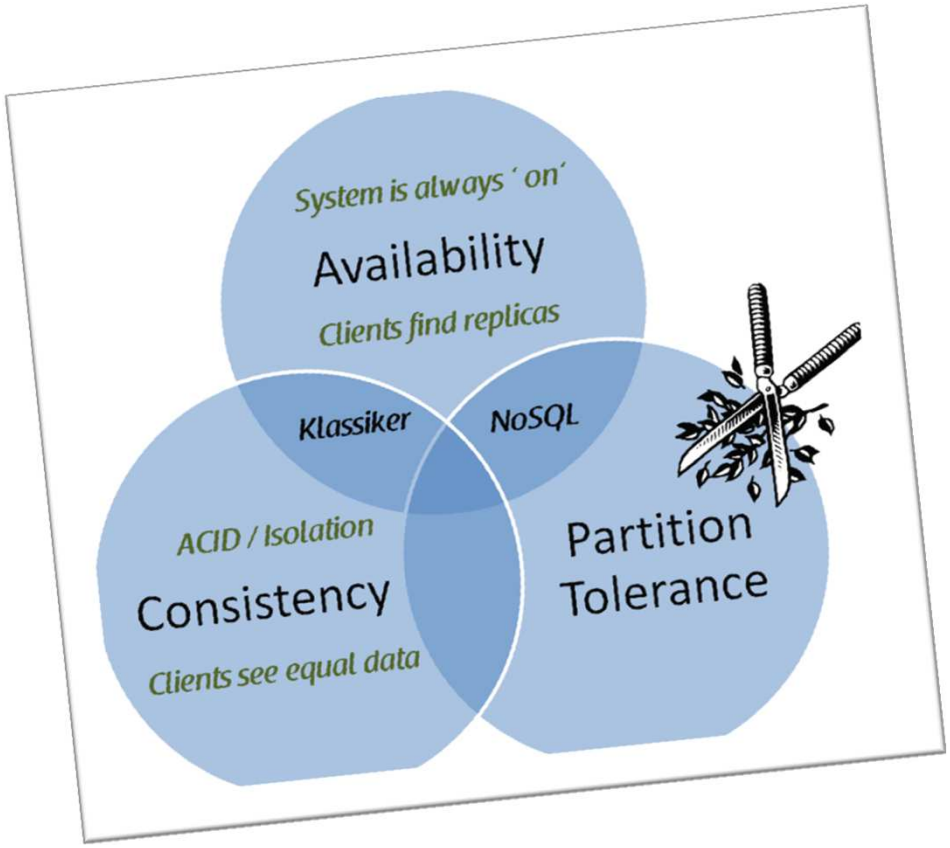
#8 up 24/7

Google 40.000$ /Min
Amazon 35.000$ /Min
Yahoo 10.000$ /Min

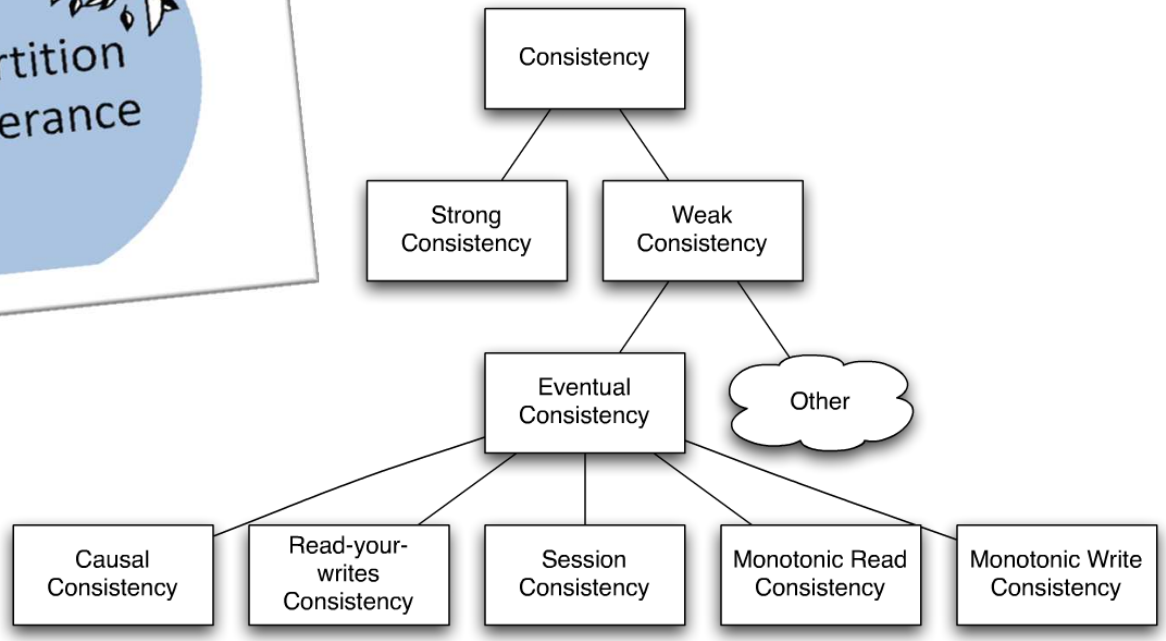
in 2008 !!

WHY #9 Programmer ease of use

WHY #10 Changing CAP needs



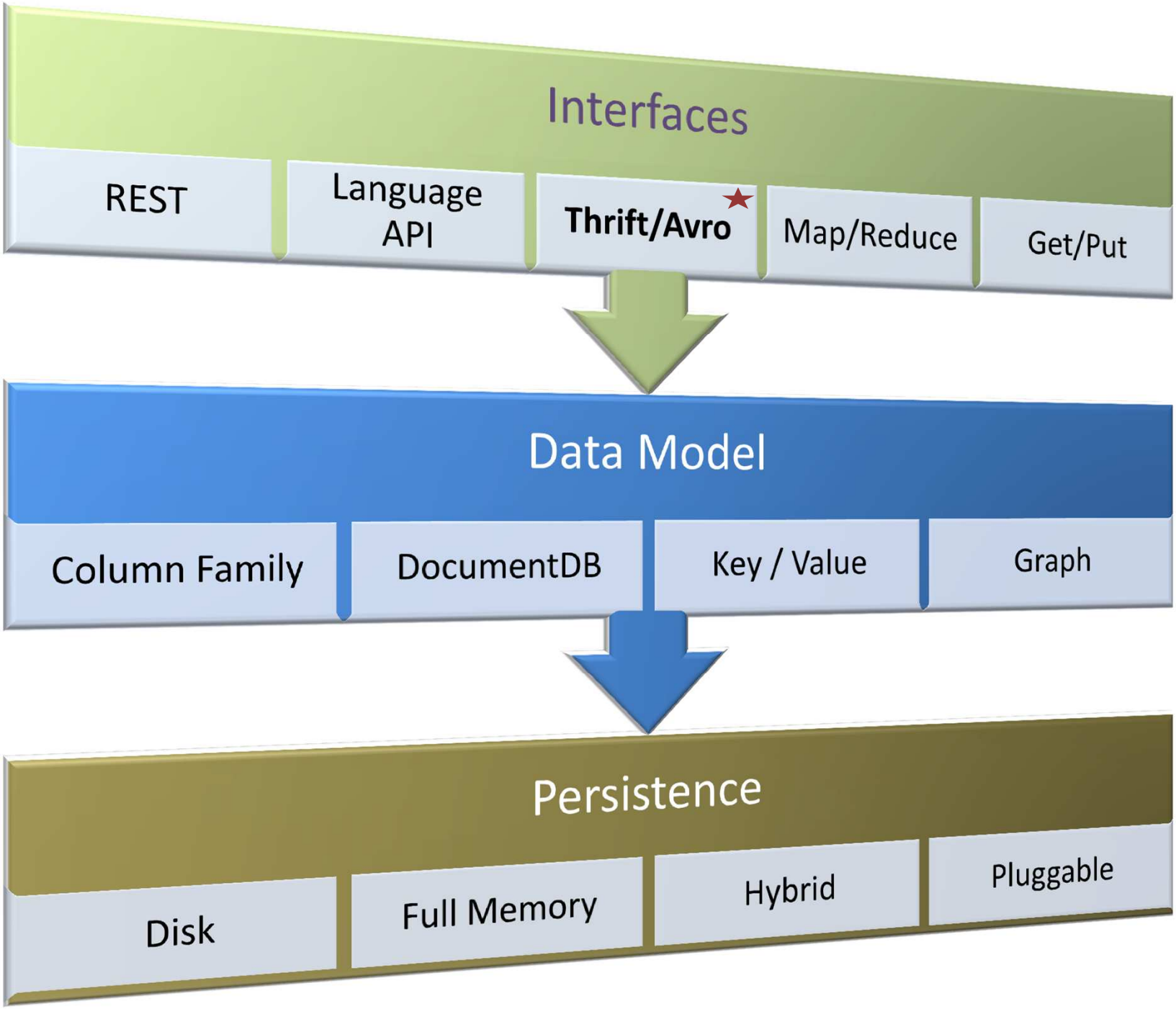
CONSISTENCY MODELS



next talk!

Relationale DB	Server	Database	Table	Primary Key			
Key Value DB	Cluster	Keyspace		Key	Value		
Column Family DB	Cluster	Table / Keyspace	Column Family	Key	Column Name	Column Value	Super Column optional
Document DB	Cluster	Docspace		Doc Name	Doc Content		
GraphDB	Server	Nodespace	Nodes & Links				

DATA MODEL

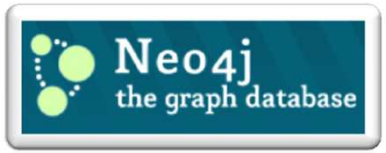

Interfaces
REST
Language API
Thrift/Avro
Map/Reduce
Get/Put
Data Model
Column Family
DocumentDB
Key / Value
Graph
Persistence
Disk
Full Memory
Hybrid
Pluggable

Column Family

DocumentDBs

Key/ValueDBs

Voldemort, Chordless, Scalaris, Dynamo / Dynomite



GraphDBs

others

db4o, Versant, Objectivity, Gemstone, Progress, Mark Logic, EMC Momentum, Tamino, GigaSpaces, Hazelcast, Terracotta, …

> 220 DBs

tough consulting…

+ Scaling= new node
+ Community
+ API
- Replication
- SetUp, Optimize, Management

+ Scaling= new node
+ Replication
+ Configuration (r, w)
- Documentation
- Query
- (storage-conf.xml)

	HBase	Cassand.	DynDB	Mongo	CouchBS	Riak	Redis	ES
schema free	Chunks	TAB	TAB	JSON	JSON	JSON	K/V	JSON
‚realtime‘								
performance	mass data			100k			100k	
scaling							V3	
ring / shardRepl	chunks	ring	invisible	ShRe	ring	ring	repl	ShRe
self tuning		In progress	SaaS	hard				
prod / tools			why?	MMS			Why?	
aggregations						?	DSS	
queries	Hive,Pig	CQL?					DSS	
full text s		?					DSS	
filemanagem								
community								
APIs				++	-		++	REST
support			why?					
docs				++	+/-			
misc				Geo, StoredProcs	Geo	persist config	limited sharding	persist config

Performance
Transactions
Queries
Architecture
non-funct-Requirements
Data

Analyse your Data

Domain-Data, Log-Data, Event-Data, Message-Data, critical Data, Business-Data, Meta-Data, temp Data, Session-Data, Geo Data, etc.

Data- / Storage-Model:

relational, column-o, doc-alike, graphs, objects, etc.

What Types / Type-System?

Data-Navigation, Data Amount, Data Komplexity (Deep XML?)

ACID vs. BASE vs. Mixture?
CAP decisions

Performance Dimension Analysis

Latency, Request behaviour, Throughput

Scale-Up vs Scale-Out

Query Requirements

Typical queries, Tools, Ad-Hoc Queries, SQL / LINQ needed, Map/Reduce? …

Distribution Architecture

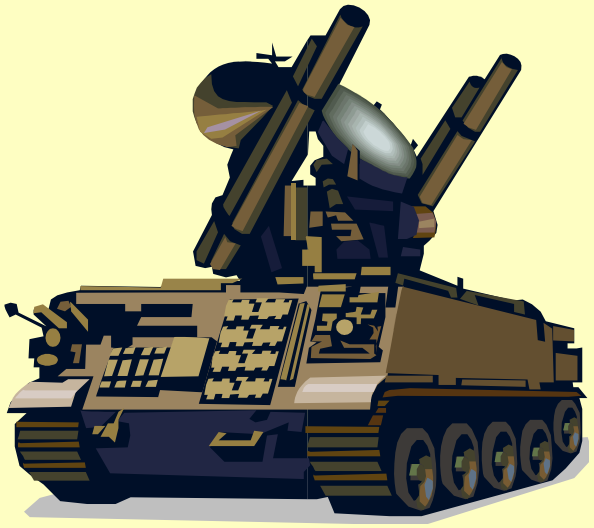
local, parallel, distributed / grid, service, cloud, mobile, p2p, …

Data Access Patterns

read / write distribution, random / sequential, Access Design Patterns

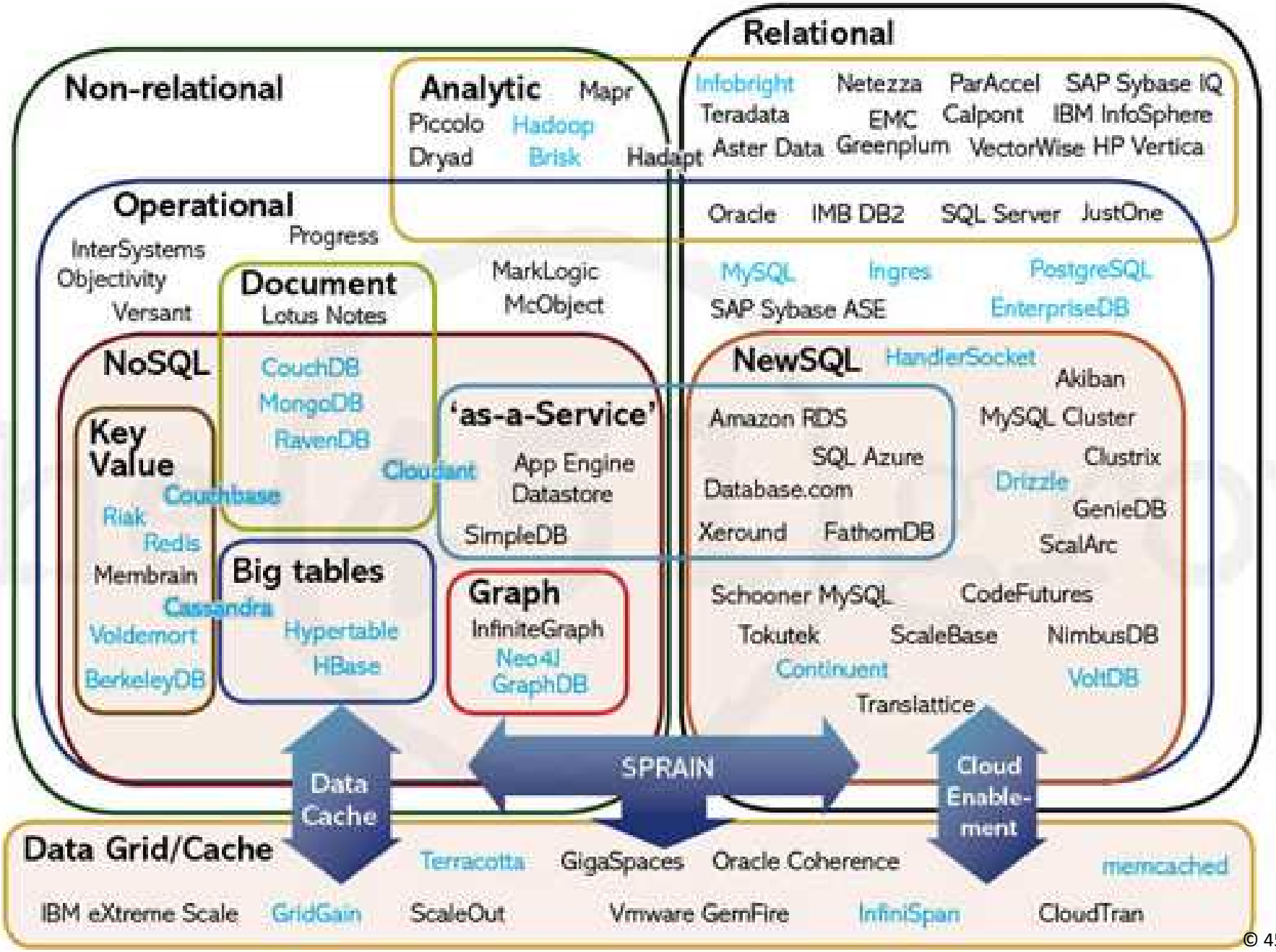
Non Functional Requirements:

Replication, Refactoring Frequency, DB-Support, Qualification / simplicity, Company restrictions, DB diversity (allowed?), Security, Safety / Backup & Restore, Crash Resistance, Licence…



MoreSQL

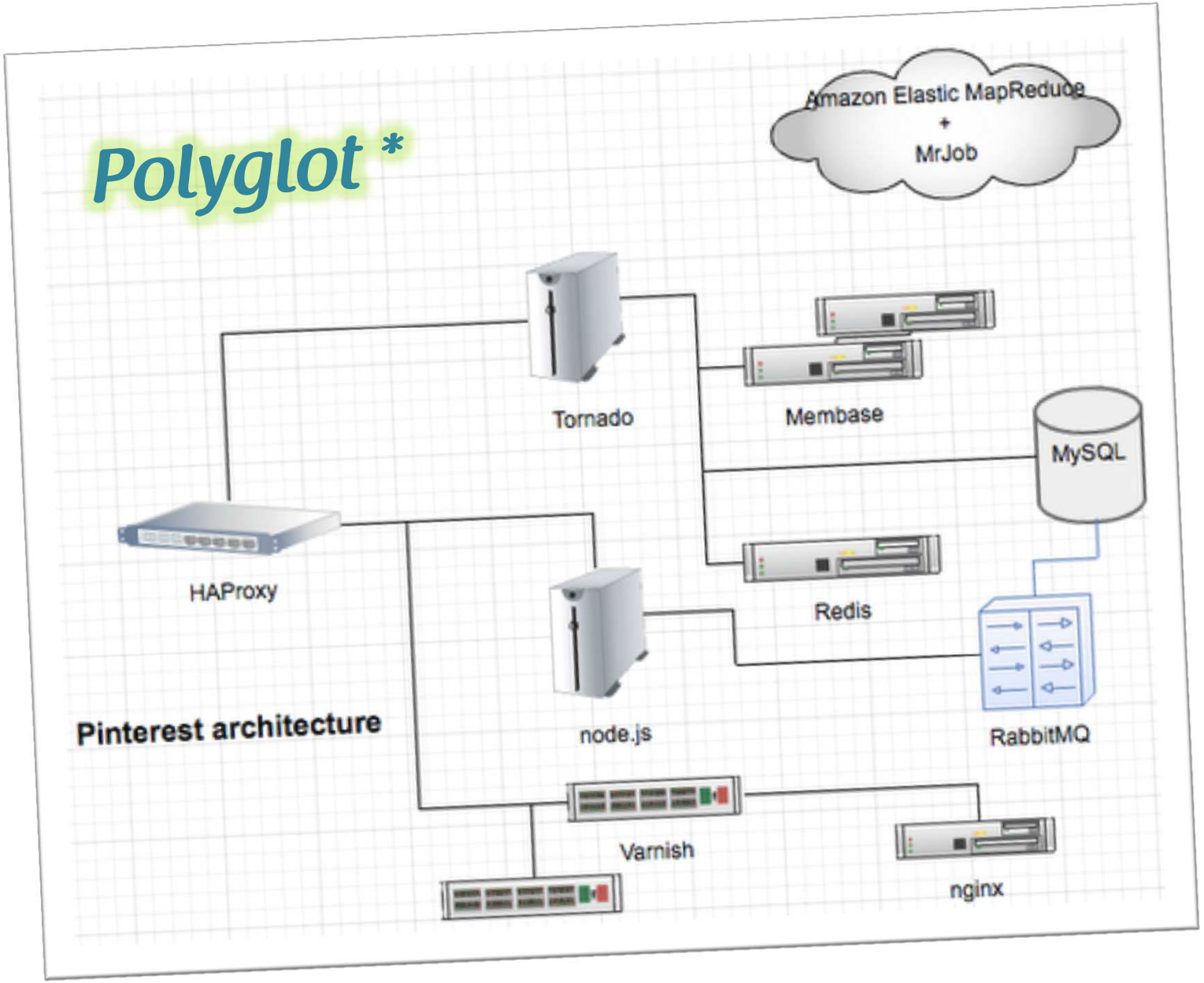
strikes back


Non-relational
Relational
Analytic
Mapr
Piccolo
Hadoop
Dryad
Brisk
Hadapt
Infobright
Netezza
ParAccel
SAP Sybase IQ
Teradata
EMC
Calpont
IBM InfoSphere
Aster Data
Greenplum
VectorWise
HP Vertica
Operational
Oracle
IMB DB2
SQL Server
JustOne
InterSystems
Progress
Objectivity
Versant
Document
Lotus Notes
MarkLogic
McObject
MySQL
Ingres
PostgreSQL
SAP Sybase ASE
EnterpriseDB
NoSQL
CouchDB
MongoDB
RavenDB
Cloudant
Key Value
Couchbase
Riak
Redis
Membrain
Cassandra
Voldemort
BerkeleyDB
Big tables
Hypertable
HBase
'as-a-Service'
App Engine Datastore
SimpleDB
Graph
InfiniteGraph
Neo4J
GraphDB
NewSQL
HandlerSocket
Akiban
Amazon RDS
MySQL Cluster
SQL Azure
Clustrix
Drizzle
Database.com
GenieDB
Xeround
FathomDB
ScalArc
Schooner MySQL
CodeFutures
Tokutek
ScaleBase
NimbusDB
Continuent
VoltDB
Translattice
Data Cache
SPRAIN
Cloud Enable-ment
Data Grid/Cache
Terracotta
GigaSpaces
Oracle Coherence
memcached
IBM eXtreme Scale
GridGain
ScaleOut
Vmware GemFire
InfiniSpan
CloudTran

© 451 G

Check hybrid solutions!

easier & better then memcache + RDBMS



© Alex Popescu

Conclusion #1

There is no
“one perfect solution”

Check hybrid solutions
and NewSQL DBs too!

Conclusion #2

Conclusion #3

why

THANKS!

Q&A

please contact
edlich.de