

NoSQL

A apresentação já vai começar...

NoSQL

Uma “nova” visão sobre a persistência de dados

Eder Ignatowicz

Eder Ignatowicz...

FACCAMP
Faculdade Campo Limpo Paulista

InfoQ
news



github /ederig
SOCIAL CODING



NoSQL



@ederign

<http://jogano10.com>

NoSQL

Abaixo ao Banco de Dados Relacional



@ederign

<http://jogano10.com>

NoSQL

*Abaixo ao Banco de Dados Relacional
como bala de PRATA!*



@ederign

<http://jogano10.com>

Falando sério...



@ederign

<http://jogano10.com>

No SQL

Not Only SQL



@ederign

<http://jogano10.com>

No SQL

*Not **O**nly SQL*



@ederign

<http://jogano10.com>

No SQL

Not Only SQL

*Uma **nova** escola*



@ederign

<http://jogano10.com>

Mas professor? Qual o problema com o Banco de Dados Relacional?



@ederign

<http://jogano10.com>

Bancos de Dados Relacionais



@ederign

<http://jogano10.com>

Bancos de Dados Relacionais

Solução madura



@ederign

<http://jogano10.com>

Bancos de Dados Relacionais

A melhor solução para 90,02%* dos
teus problemas de persistência



@ederign

* Instituto Eder de Estatística

<http://jogano10.com>



Bancos de Dados Relacionais

ORACLE®

PostgreSQL



@ederign

<http://jogano10.com>



@ederign

<http://jogano10.com>

Sistemas Distribuídos



@ederign

<http://jogano10.com>

Sistemas Distribuídos

Componentes computacionais localizados
em regiões físicas distintas que trocam
mensagens



@ederign

<http://jogano10.com>

Sistemas Distribuídos

Possuem três características desejadas:



@ederign

<http://jogano10.com>

Sistemas Distribuídos

Possuem três características desejadas:

Consistência: validade, precisão e integridade da informação compartilhada entre diversos nós



@ederign

<http://jogano10.com>

Sistemas Distribuídos

Possuem três características desejadas:

Disponibilidade: garantia que todas as requisições recebam uma resposta adequada, ou seja, quando uma falha ocorrer, o repositório deverá manter-se em funcionamento;



@ederign

<http://jogano10.com>

Sistemas Distribuídos

Possuem três características desejadas:

Tolerância a particionamentos: se ocorrer uma falha de rede que divida os nós em dois grupos que não conversam mais entre si, o repositório deve possibilitar acesso e processamento nos dois subgrupos.

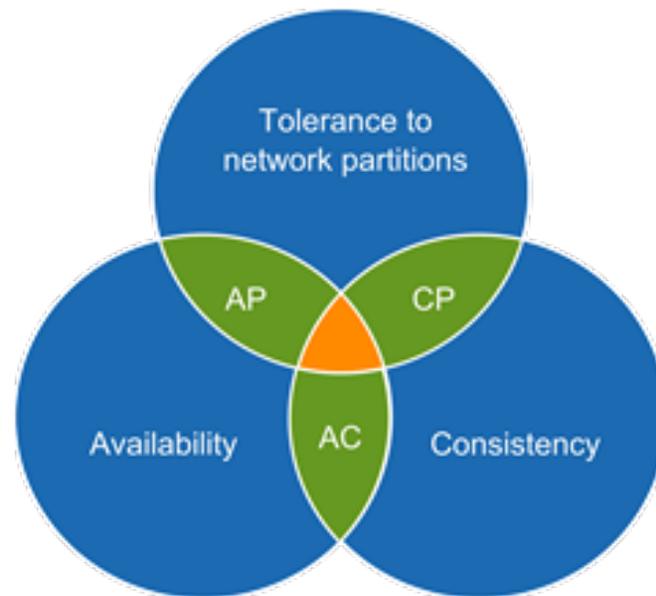


@ederign

<http://jogano10.com>

Teorema CAP

É impossível obter as três características ao mesmo tempo



@ederign

<http://jogano10.com>

Teorema CAP

Trade-off



@ederign

<http://jogano10.com>

Teorema CAP

*CA **sem** o P: Sistemas que provêem transações distribuídas, somente podem garantir este serviço se não ocorrer particionamento do servidores;*



@ederign

<http://jogano10.com>

Teorema CAP

*CP **sem** A: No caso de um particionamento, as transações serão bloqueadas até a recuperação do sistema com objetivo de evitar eventuais conflitos de dados (o que provocaria uma inconsistência);*



@ederign

<http://jogano10.com>

Teorema CAP

*AP **sem** C: É aceitado que parte dos nós eventualmente estejam em determinado momento em um estado inconsistente.*



@ederign

<http://jogano10.com>

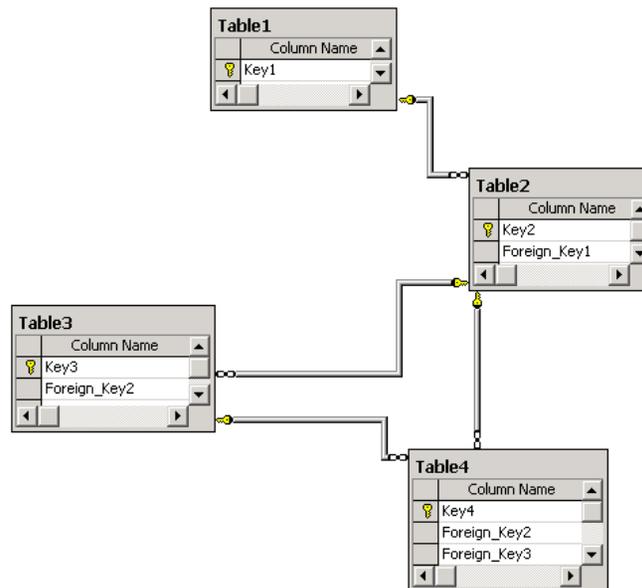
Mas professor? E o Banco Relacional com isto?



@ederign

<http://jogano10.com>

Bancos Relacionais



@ederign

<http://jogano10.com>

Bancos Relacionais

Facilitaram a vida dos desenvolvedores...



@ederign

<http://jogano10.com>

Bancos Relacionais

Facilitaram a vida dos desenvolvedores...

SGBD é responsável pela **verificação** e **garantias** de **integridade** dos dados, **controle** de **concorrência**, **recuperação** de **falhas** e **segurança** para os SGBDs



@ederign

<http://jogano10.com>

Bancos Relacionais

ACID

(Atomicity, Consistency, Isolation, Durability)



@ederign

<http://jogano10.com>

Bancos Relacionais

ACID
Atomicity

Transação deve ser executada por completo ou não ser executada



@ederign

<http://jogano10.com>

Bancos Relacionais

ACID
Consistency

Toda transação deve garantir a integridade do banco de dados



@ederign

<http://jogano10.com>

Bancos Relacionais

ACID
Isolation

Mesmo que as ações de diversas transações sejam executadas de forma intercalada, o resultado dessas execuções deve ser idêntico



@ederign

<http://jogano10.com>

Bancos Relacionais

ACID Durability

SGBD relacional deve assegurar que o efeito de transações persista, mesmo que o sistema falhe.



@ederign

<http://jogano10.com>

Bancos Relacionais

Ou em Sistemas Distribuídos...

Durability trata de garantir que dados que foram “comitados” sejam persistentes em uma única máquina e que esses dados sejam replicados em um determinado número de máquinas ou em diferentes centros de dados.



@ederign

<http://jogano10.com>

Bancos Relacionais

Consistência Absoluta

Isto implica em detrimento da tolerância a partionamentos em favor da consistência e disponibilidade (CA sem o P)

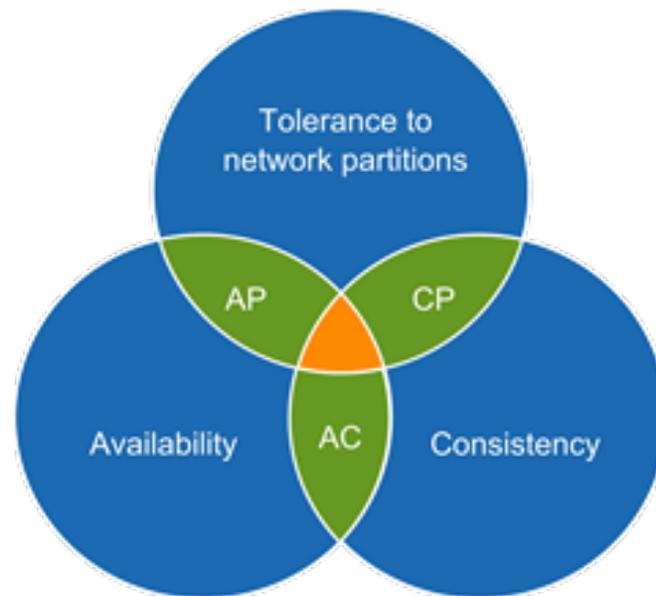


@ederign

<http://jogano10.com>

Teorema CAP

É impossível obter as três características ao mesmo tempo



@ederign

<http://jogano10.com>

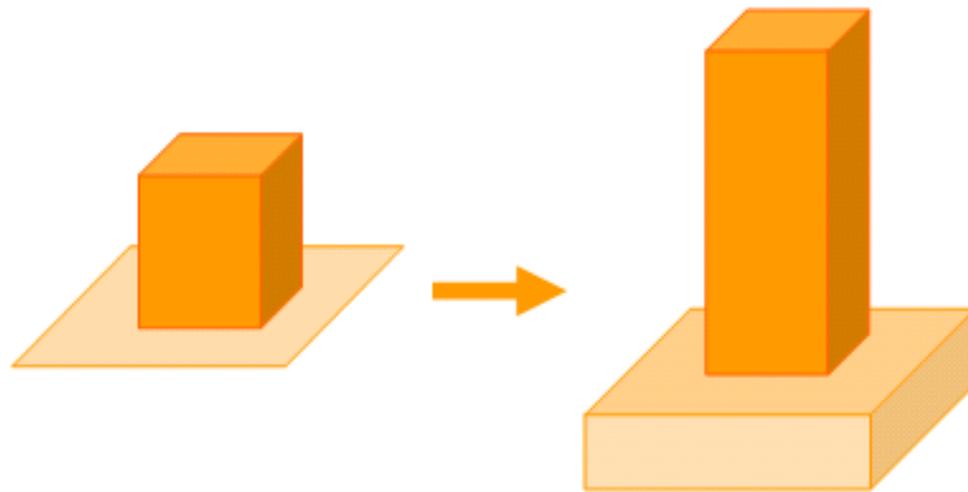
Mas professor? O que acontece se eu não tenho tolerância a particionamentos?



@ederign

<http://jogano10.com>

Escalabilidade Vertical



@ederign

<http://jogano10.com>

Ah professor, ai que estão
90,02% dos sistemas



@ederign

<http://jogano10.com>

Mas...



@ederign

<http://jogano10.com>

Evolução da Internet



twitter



Google™



@ederign

<http://jogano10.com>

Era dos Petabytes

WHAT IS A PETABYTE?

TO UNDERSTAND A PETABYTE WE
MUST FIRST UNDERSTAND A
GIGABYTE.

| | |
|-------------------------|--|
| 1 GIGABYTE | ▪ 7 MINUTES OF HD-TV VIDEO |
| 2 GIGABYTES | ▪ 20 YARDS OF BOOKS ON A SHELF |
| 4.7 GIGABYTES | ▪ SIZE OF A STANDARD DVD-R |

THERE ARE A MILLION GIGABYTES
IN A PETABYTE



@ederign

<http://jogano10.com>

A **PETABYTE** IS A LOT OF DATA

1

PETABYTE



20 MILLION
FOUR-DRAWER FILING CABINETS
FILLED WITH TEXT

1

PETABYTE



13.3 YEARS
OF HD-TV VIDEO

1.5

PETABYTES



SIZE OF THE 10 BILLION
PHOTOS ON **FACEBOOK**

20

PETABYTES



THE AMOUNT OF DATA **PER**
PROCESSED BY **GOOGLE** **DAY**



@ederign

<http://jogano10.com>



***20 PETABYTES/
DIA***

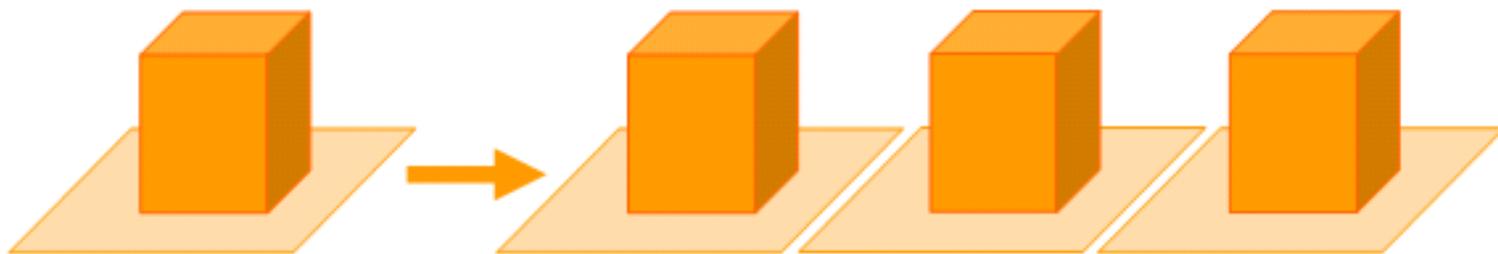


@ederign

<http://jogano10.com>

Escalabilidade Horizontal

é Pré-Requisito

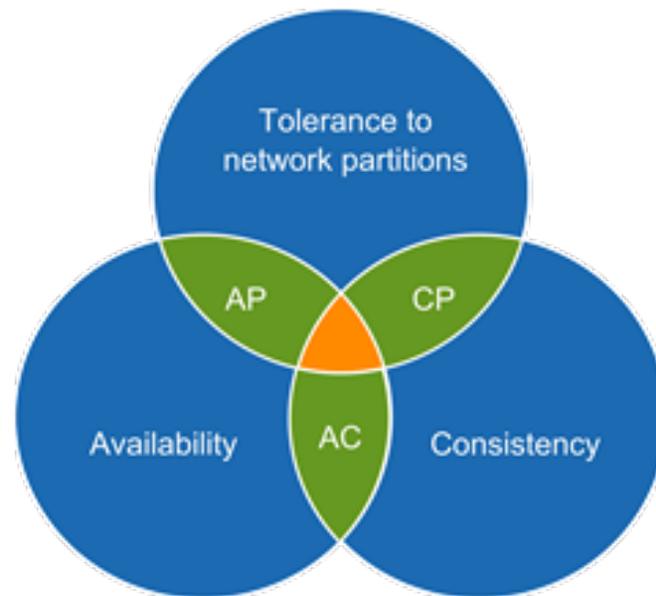


@ederign

<http://jogano10.com>

Teorema CAP

Preciso ter tolerância a partionamentos

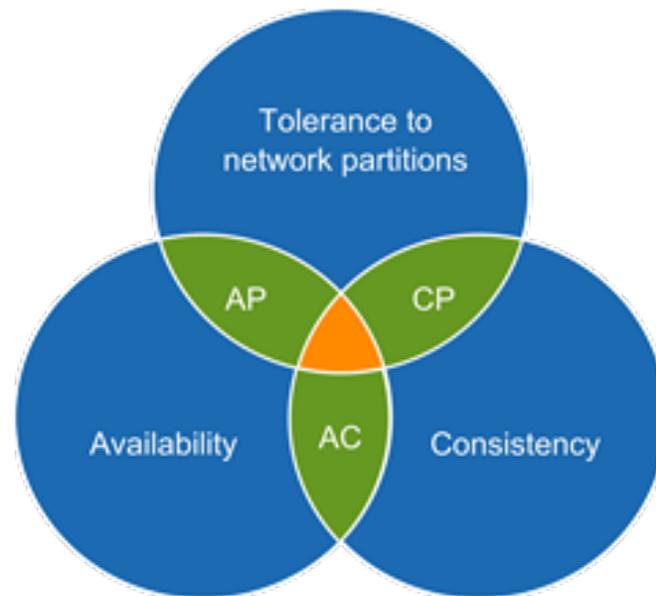


@ederign

<http://jogano10.com>

Teorema CAP

Preciso exercitar o tradeoff de maneiras diferentes



@ederign

<http://jogano10.com>

Bancos Relacionais

Outros problemas...

Como unir tabelas (join) em um sistema distribuído?



@ederign

<http://jogano10.com>

Bancos Relacionais

*Capacidade de lidar com diferenças sintáticas,
estruturais e semânticas dos dados*

*Todos os dados necessitam serem convertidos
em tabelas.*

*Este modelo convertido é por vezes complexo
e ineficiente.*



@ederign

<http://jogano10.com>

Bancos Relacionais

SQL só é eficiente para dados estruturados



@ederign

<http://jogano10.com>

Not
Only **SQL**



@ederign

<http://jogano10.com>

Novo Paradigma

BASE

(Basically Available, Soft state, Eventual consistency)



@ederign

<http://jogano10.com>

Novo Paradigma

BASE

Eventual consistency



@ederign

<http://jogano10.com>

Novo Paradigma

BASE

Em um sistema distribuído é fato que não podemos conseguir consistência absoluta em sistemas distribuídos escaláveis



@ederign

<http://jogano10.com>

Not
Only SQL

Estrutura de dados



@ederign

<http://jogano10.com>

Not only SQL

Desnormalização



@ederign

<http://jogano10.com>

Not Only SQL

Membros da Família

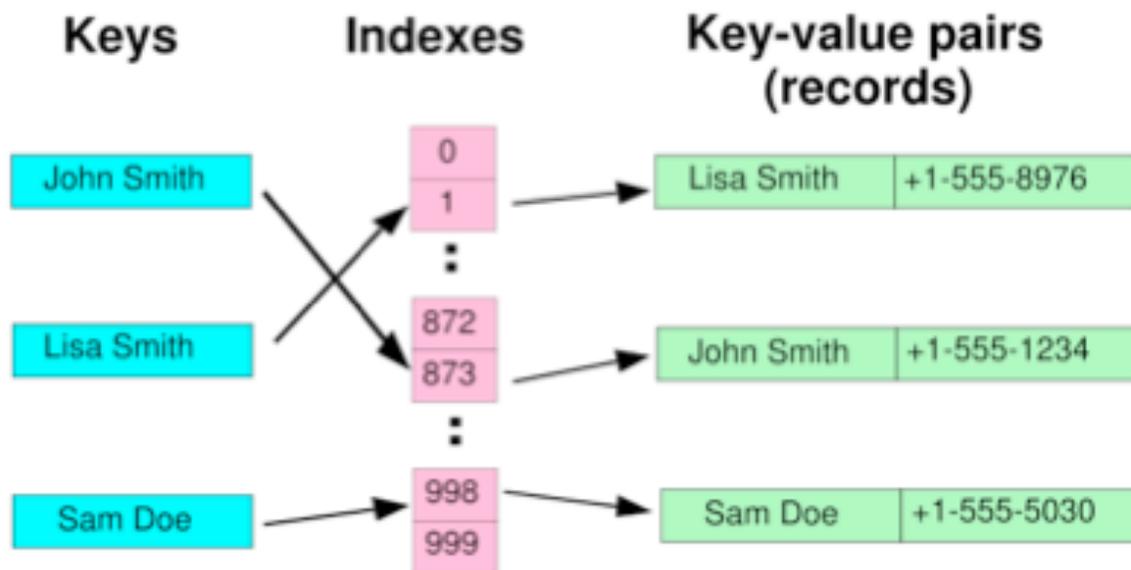


@ederign

<http://jogano10.com>

No SQL

Key-value stores



@ederign

<http://jogano10.com>

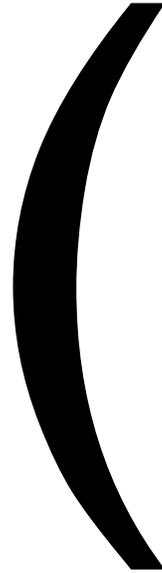
No SQL

Key-value stores



@ederign

<http://jogano10.com>



@ederign

<http://jogano10.com>

RESTful Web Services

REpresentational **S**tate **T**ransfer (REST)

Baseia-se e utiliza os métodos HTTP

Interface Uniforme e Padrão



@ederign

<http://jogano10.com>

GET

- Buscar recursos
- Cache

POST

- Criar um novo recurso

PUT

- Atualizar um recurso existente

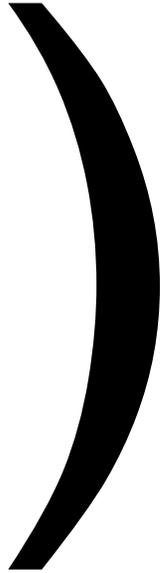
DELETE

- Remover um recurso



@ederign

<http://jogano10.com>



@ederign

<http://jogano10.com>



```
$ curl -i -X POST http://localhost:8091/riak/animals \  
-H "Content-Type: application/json" \  
-d '{"nickname" : "Sergeant Stubby", "breed" : "Terrier"}'
```

```
HTTP/1.1 201 Created  
Vary: Accept-Encoding  
Server: MochiWeb/1.1 WebMachine/1.7.3 (participate in the frantic)  
Location: /riak/animals/6VZc2o7zKxq2B34kJrm1S0ma3P0  
Date: Tue, 05 Apr 2011 07:45:33 GMT  
Content-Type: application/json  
Content-Length: 0
```



@ederign

<http://jogano10.com>



```
$ curl http://localhost:8091/riak/animals/6VZc2o7zKxq2B34kJrm1S0ma3P0
```

```
$ curl -i -X DELETE http://localhost:8091/riak/animals/6VZc2o7zKxq2B34kJrm1S0ma3P0
```



@ederign

<http://jogano10.com>

No SQL

BigTable Clones (Column Family)

| Row Key | Time Stamp | Column Family : Basic | | | | Column Family : History | Column Family : Education | | | Column Family : Address | | | | |
|---------|------------|-----------------------|----------|-------|------------|-------------------------|---------------------------|-----------|---------|-------------------------|--------|------|-------|-------|
| | | Name | DOB | Photo | Start Date | Job Title | High School | Bachelor | Masters | Door | Street | City | State | Zip |
| 1 | T1 | John | DD/MM/YY | YY | DD/MM/YY | | | | | | | | | |
| 1 | T3 | | | | | Sr. System Analyst | | | | | | | | |
| 1 | T2 | | | | | Analyst | | | | | | | | |
| 1 | T2 | | | | | Programmer | | | | | | | | |
| 1 | T1 | | | | | | Science | | | | | | | |
| 1 | T1 | | | | | | | Computers | | | | | | |
| 1 | T1 | | | | | | | | | XX | XX | XX | XX | 53353 |



@ederign

<http://jogano10.com>

No SQL

BigTable Clones (Column Family)



@ederign

<http://jogano10.com>

No SQL

BigTable Clones (Column Family)

| | row keys | column family "color" | column family "shape" |
|-----|----------|---|----------------------------------|
| row | "first" | "red": "#F00" "blue": "#00F" "yellow": "#FF0" | "square": "4" |
| row | "second" | | "triangle": "3" "square": "4" |



@ederign

<http://jogano10.com>

No SQL

Document Databases

```
{  
  "nickname" : "kimchy",  
  "name" : {  
    "firstName" : "Shay",  
    "lastName" : "Bannon"  
  },  
  "birthdate" : "1977-11-15",  
  "projects" : [  
    "compass",  
    "elasticsearch"  
  ],  
  "wowLevel" : 3,  
  "wowLevelFine" : 3.14  
}
```

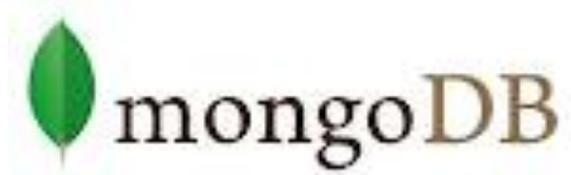


@ederign

<http://jogano10.com>

No SQL

Document Databases



@ederign

<http://jogano10.com>

No SQL

Document Databases

```
$ curl http://localhost:5984/music/  
{  
  "db_name": "music",  
  "doc_count": 1,  
  "doc_del_count": 0,  
  "update_seq": 4,  
  
  "purge_seq": 0,  
  "compact_running": false,  
  "disk_size": 16473,  
  "instance_start_time": "1326845777510067",  
  "disk_format_version": 5,  
  "committed_update_seq": 4  
}
```



@ederign

<http://jogano10.com>

No SQL

Document Databases

```
$ curl -i -X PUT \  
  "http://localhost:5984/music/74c7a8d2a8548c8b97da748f43000f1b" \  
  -H "Content-Type: application/json" \  
  -d '{  
    "_id": "74c7a8d2a8548c8b97da748f43000f1b",  
    "_rev": "1-2fe1dd1911153eb9df8460747dfe75a0",  
    "name": "Wings",  
    "albums": ["Wild Life", "Band on the Run", "London Town"]  
  }'  
HTTP/1.1 201 Created  
Server: CouchDB/1.1.1 (Erlang OTP/R14B03)  
Location: http://localhost:5984/music/74c7a8d2a8548c8b97da748f43000f1b  
Etag: "2-17e4ce41cd33d6a38f04a8452d5a860b"  
Date: Wed, 18 Jan 2012 00:43:39 GMT  
Content-Type: text/plain;charset=utf-8  
Content-Length: 95  
Cache-Control: must-revalidate  
  
{  
  "ok":true,  
  "id":"74c7a8d2a8548c8b97da748f43000f1b",  
  "rev":"2-17e4ce41cd33d6a38f04a8452d5a860b"  
}
```



@eder }

jogano10.com

No SQL

Document Databases

```
$ curl http://localhost:5984/music/_design/artists/_view/by_name?\  
limit=5&startkey=%22C%22  
{ "total_rows": 100, "offset": 16, "rows": [  
  { "id": "340296", "key": "Calex B", "value": "340296" },  
  { "id": "353888", "key": "carsten may", "value": "353888" },  
  { "id": "272", "key": "Chroma", "value": "272" },  
  { "id": "351138", "key": "Compartir D\u00f3na Gustet", "value": "351138" },  
  { "id": "364714", "key": "Daringer", "value": "364714" }  
]
```

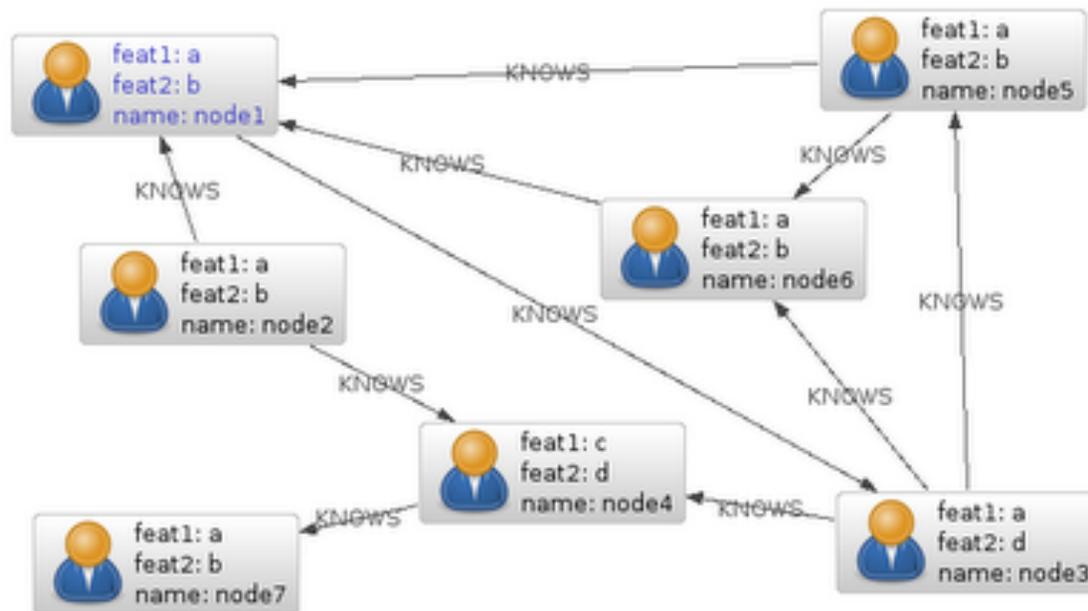


@ederign

<http://jogano10.com>

No SQL

Graph Databases (Social)



@ederign

<http://jogano10.com>

No SQL

Graph Databases (Social)



@ederign

<http://jogano10.com>

No SQL

Graph Databases (Social)

```
gremlin> spatlese = g.addVertex([name : 'Prancing Wolf Spatlese 2007'])  
==> v[5]  
gremlin> g.addEdge(pwolf, spatlese, 'produced')  
==> e[4][3-produced->5]
```



@ederign

<http://jogano10.com>

No SQL

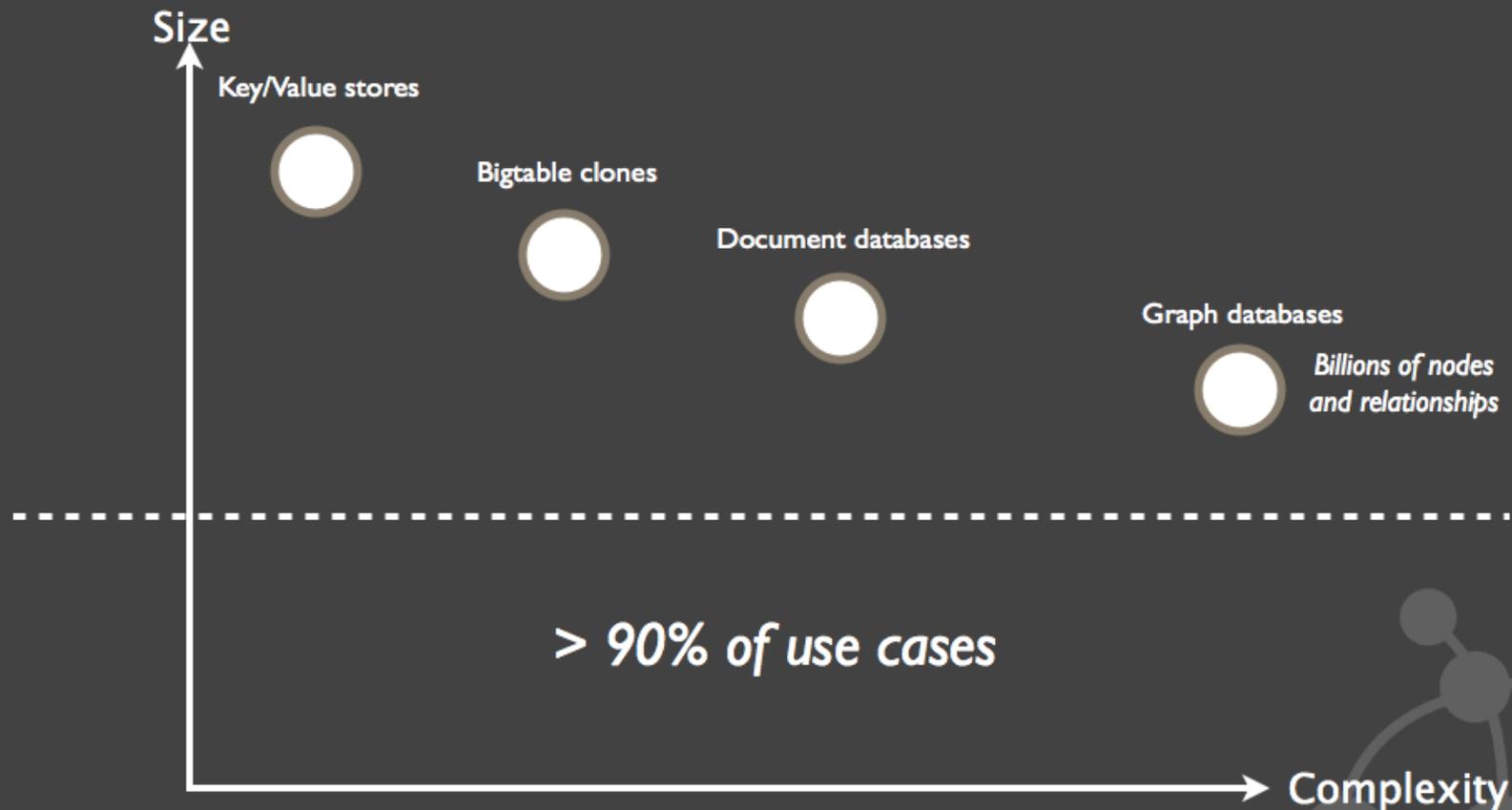
Visão Geral

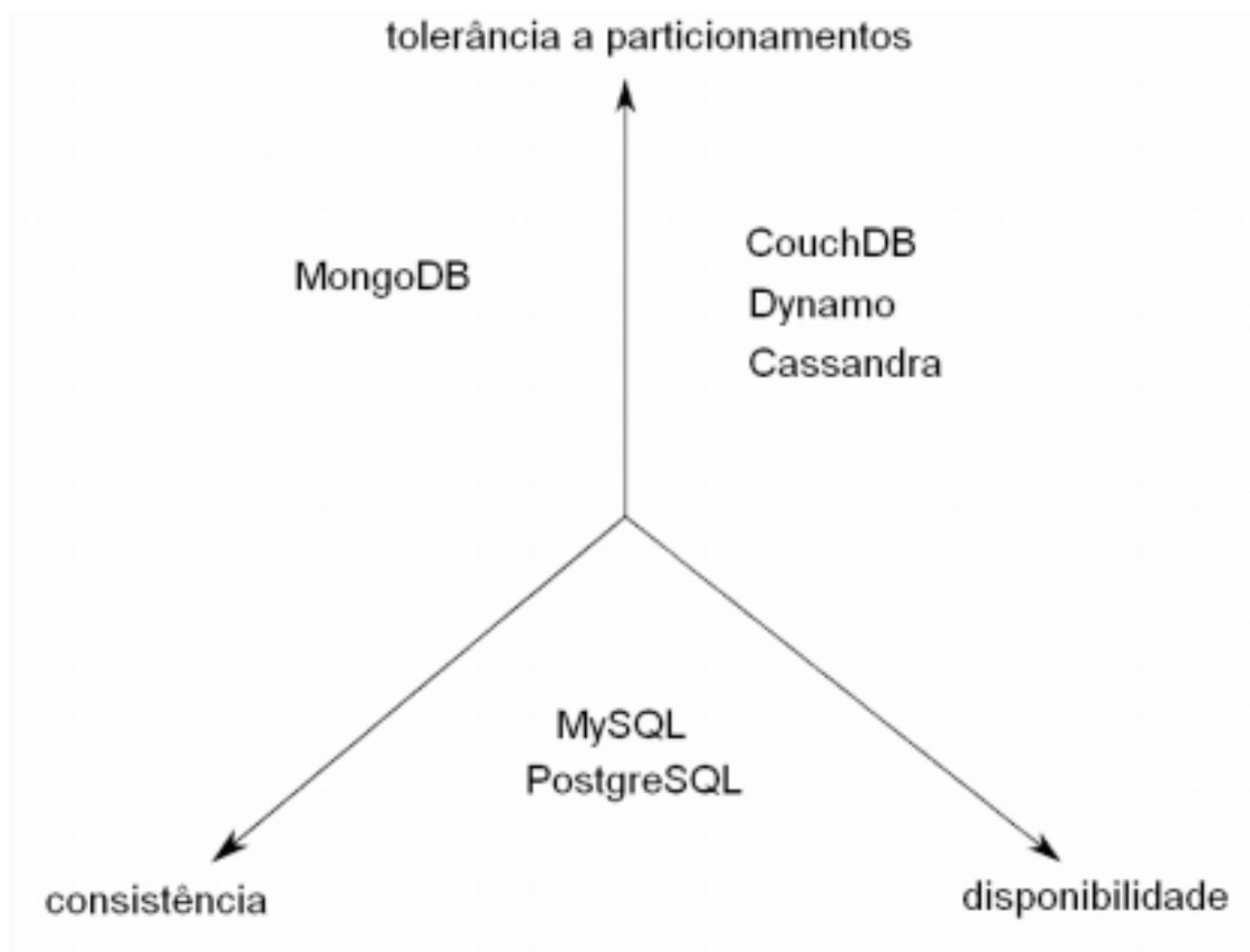


@ederign

<http://jogano10.com>

Scaling to size vs. Scaling to complexity





@ederign

<http://jogano10.com>

SQL

padrões

durabilidade

dsl comum

triggers

chave estrangeira

chave composta

transação

2 phase commit

relacional

NoSQL

flexibilidade

grafos

k-v store

documento

escalabilidade

commodity hard.

RAM

distributed

api`s



@ederign

<http://jogano10.com>

SQL

padrões

durabilidade

dsl comum

triggers

chave estrangeira

chave composta

transação

2 phase commit

relacional

NoSQL

flexibilidade

grafos

k-v store

documento

escalabilidade

commodity hard.

RAM

distributed

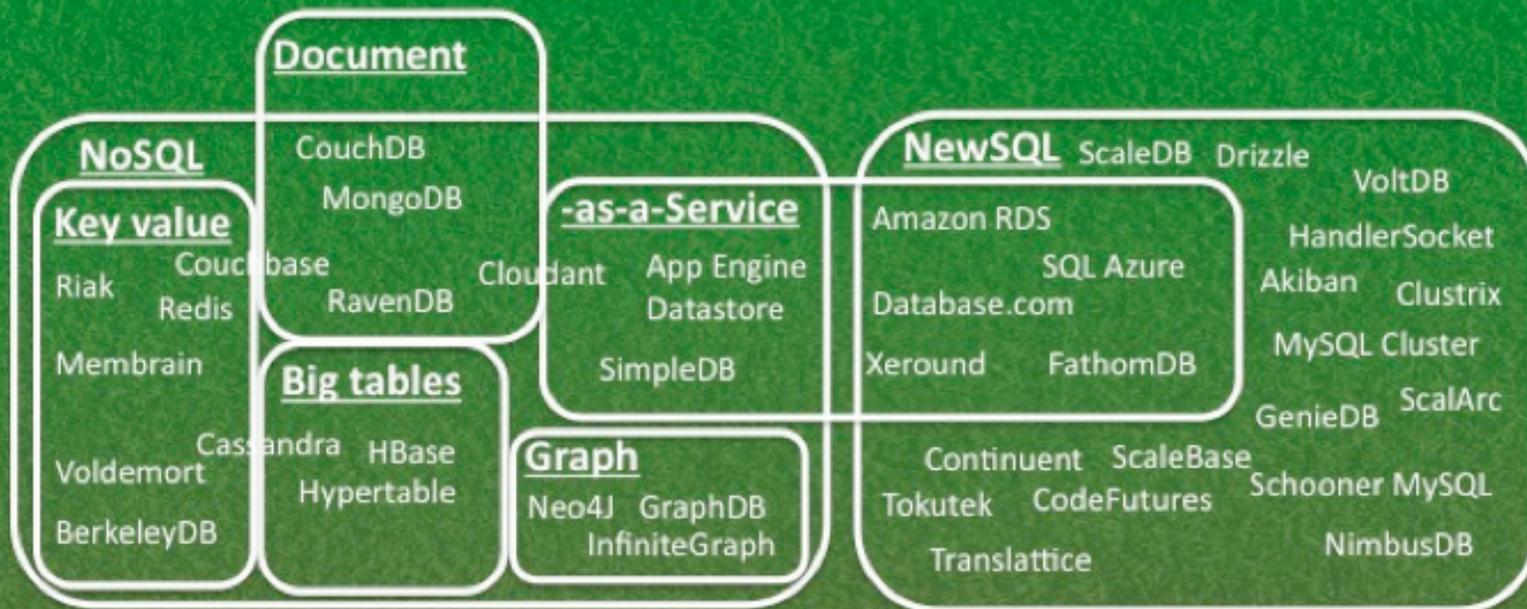
api's



@ederign

<http://jogano10.com>

NoSQL, NewSQL and Beyond





@ederign

<http://jogano10.com>