

NoSQL : MongoDB

Alexandre GINGEMBRE
Mathieu MOUROT



Sommaire

- NoSQL
- JSON
- MongoDB

NoSQL

Avant



NoSQL : avant

Avant, besoin de stocker beaucoup de données



IBM System X3500

NoSQL : avant

Beaucoup plus de données ?



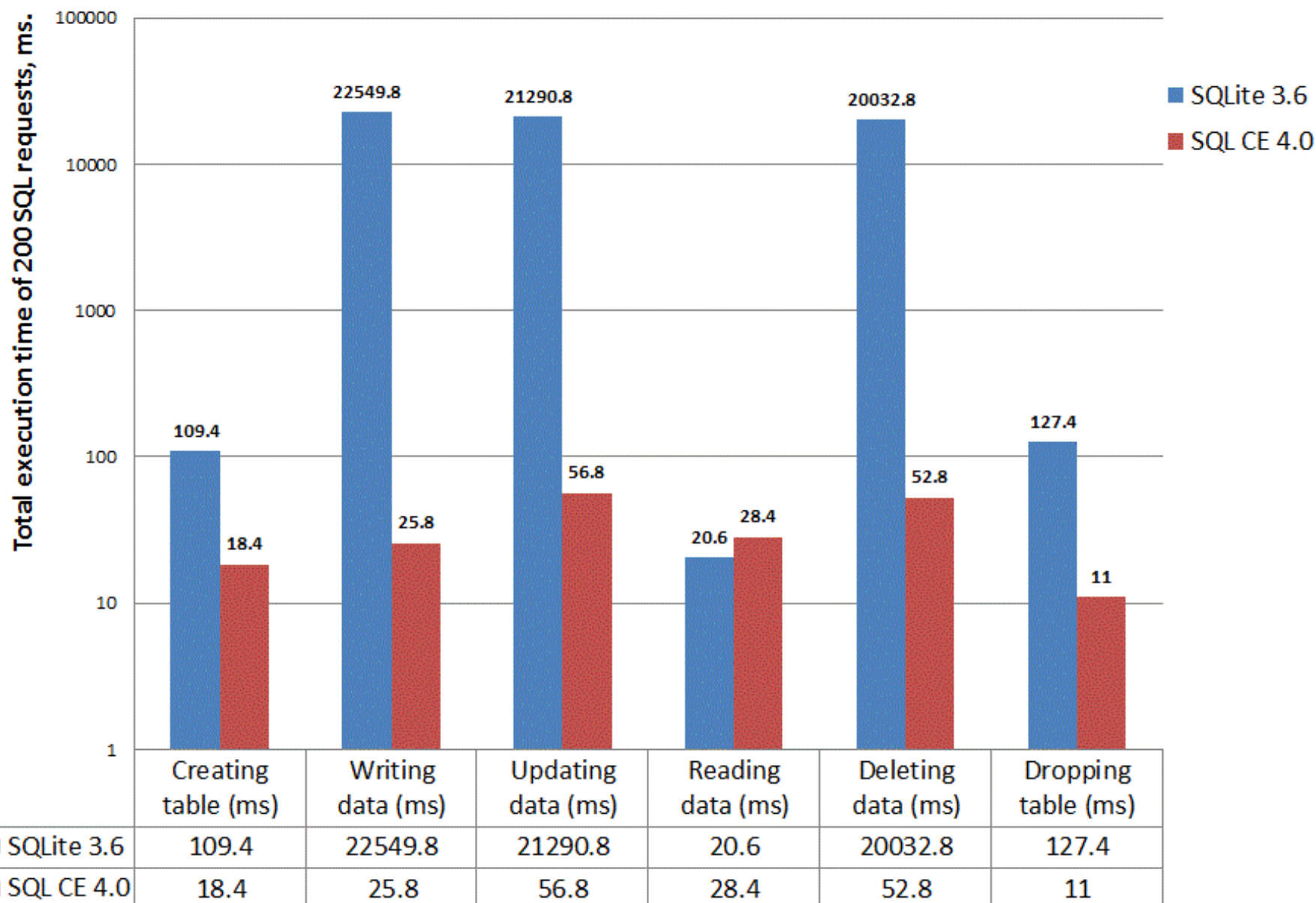
IBM zEC12

Limite SGBDR

Très vite limité :

- Base de données relationnelle, pas fait pour autant de données

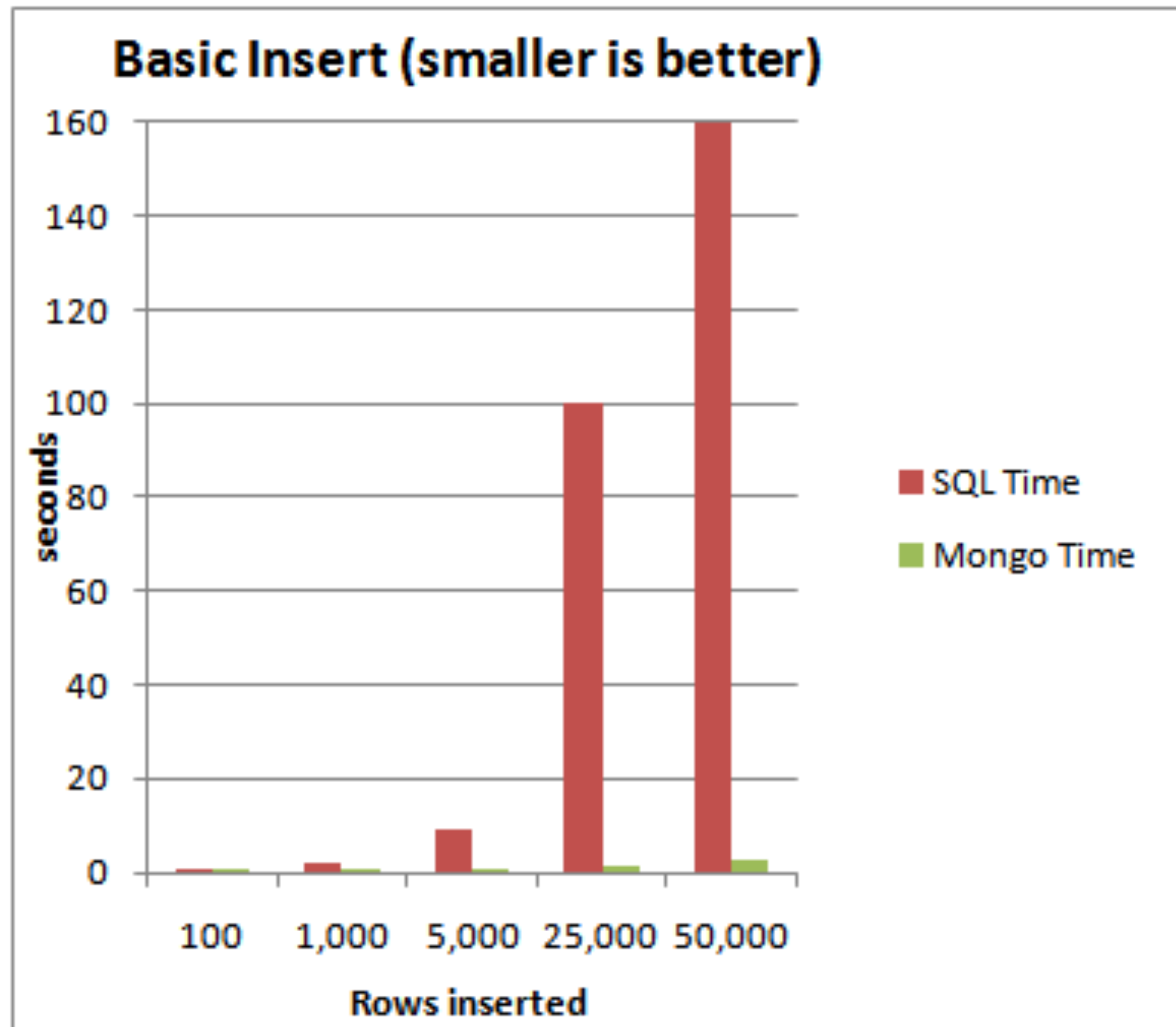
SQLite 3.6 vs. SQL CE 4.0 data access benchmarks



Averaged data for 8 experimental trials.

Crédit : Code Project

Limite SGBDR



Crédit : Michael Kennedy

NoSQL

- Not Only SQL
- Émerge début des années 2000
- Pionniers : Google (BigTable), LinkedIn (Voldemort) et Amazon (Dynamo)
- Formalisé le 11 Juin 2009 lors d'un meetup à San Francisco

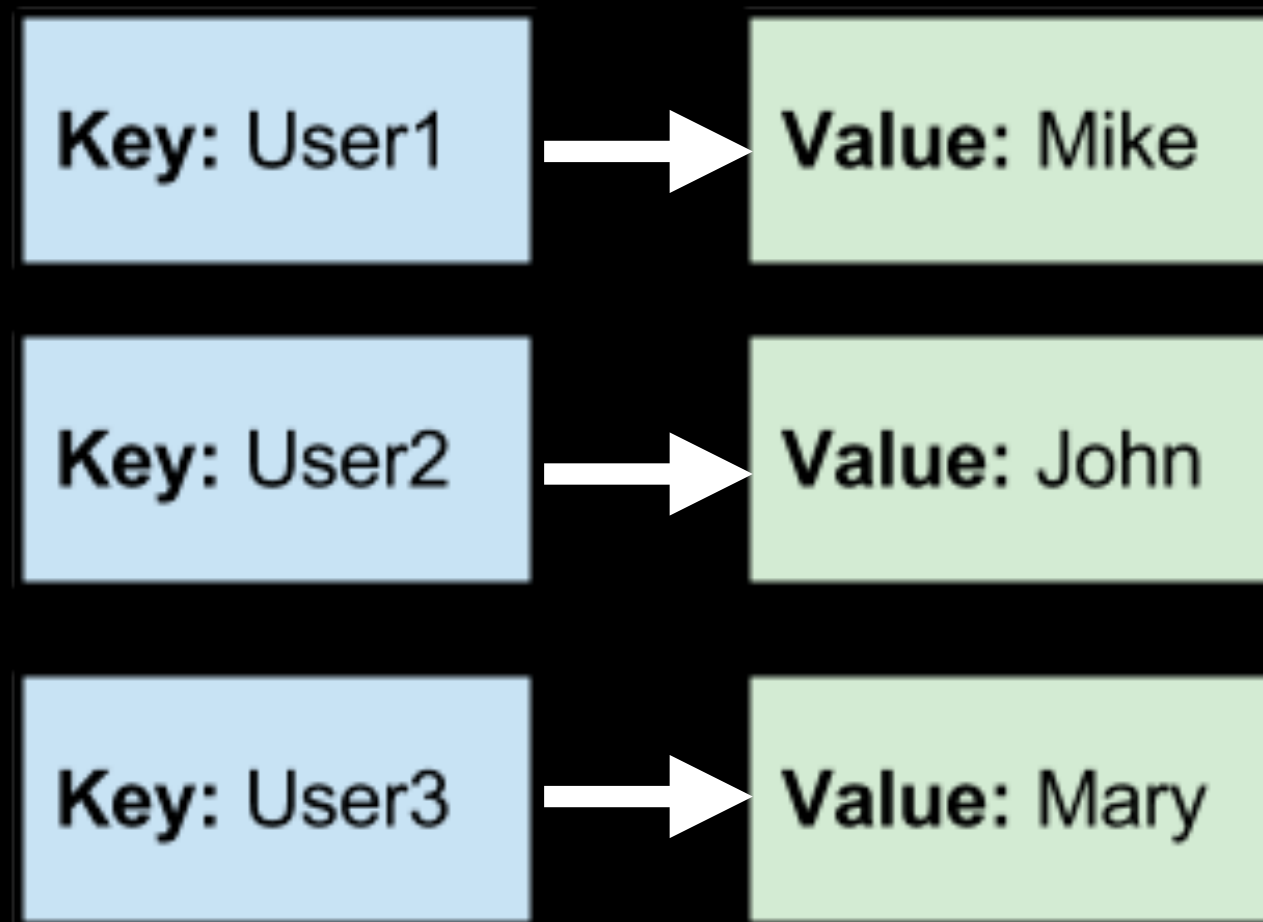
NoSQL

- 4 familles de Base de Données NoSQL :
 - Systèmes Clé-Valeur ;
 - Systèmes orientés colonnes ;
 - Graphes ;
 - Orientés Documents.

NoSQL : Clé-Valeur

- « HashMap distribué »
- Valeur : simple chaine, objet sérialisé, ...
- Applicatif qui possède l'intelligence

NoSQL : Clé-Valeur



Crédit : FWoelffel

NoSQL : Clé-Valeur



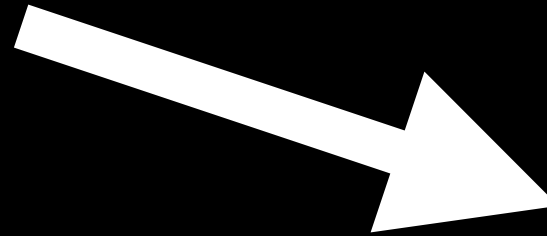
etc.

NoSQL : Orienté Colonne

- Comme table SGBDR, sauf que nombre colonne dynamique
- Nombre évolue suivant l'enregistrement
- Évite d'avoir des NULL

NoSQL : Orienté Colonne

Chaque ID correspond
une liste de couple clé-valeur



	A	B	C	D	E
1	foo	bar	hello		
2		Tom			
3			java	scala	cobol

Organisation d'une table dans
une BDD relationnelle

1	<div><div>A foo</div><div>B bar</div><div>C hello</div></div>				
2	<div><div>B Tom</div></div>				
3	<div><div>C java</div><div>D scala</div><div>E cobol</div></div>				

Organisation d'une table dans
une BDD orientée colonnes

Crédit : Xebia

NoSQL : Orienté Colonne

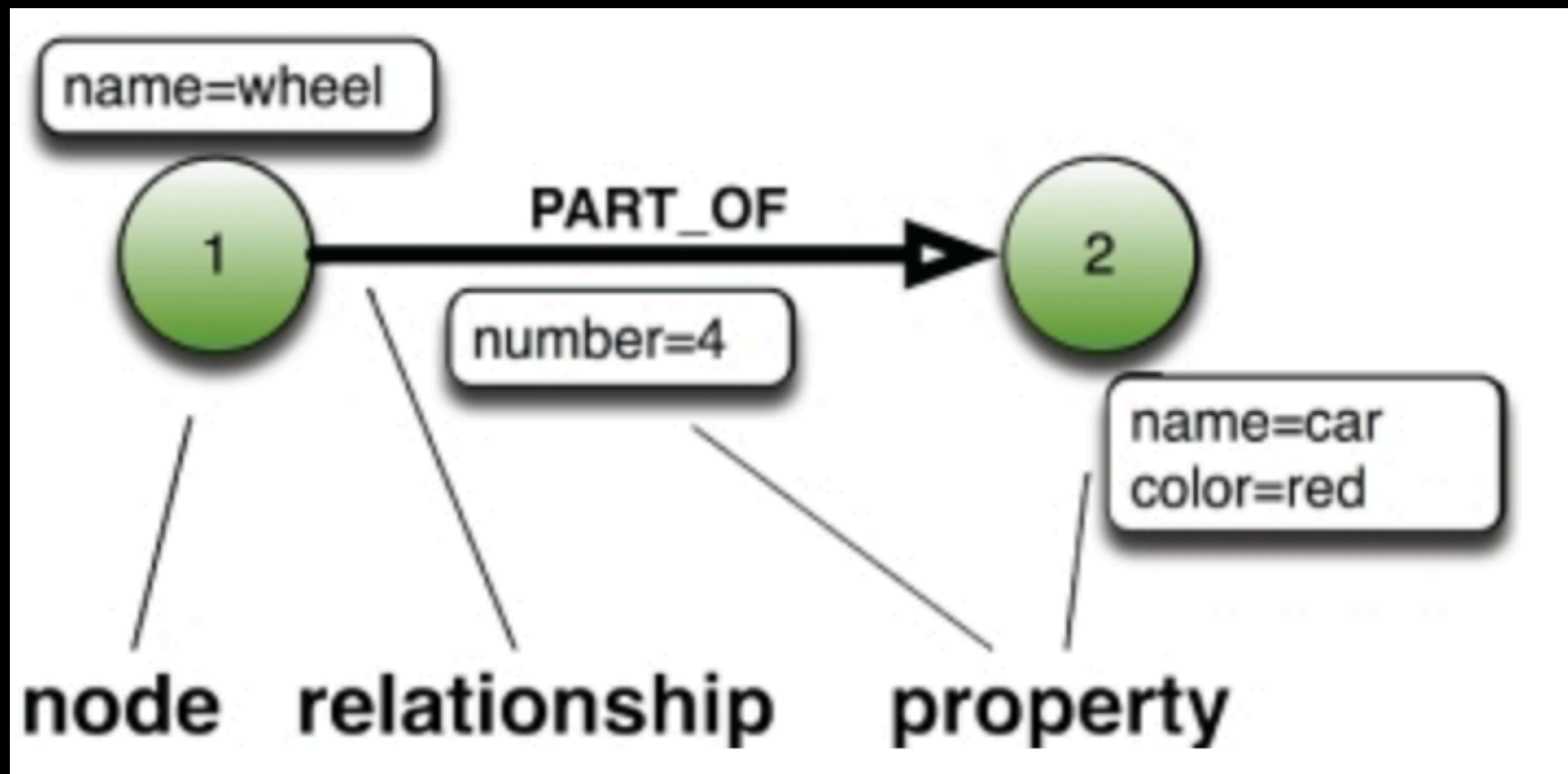


etc.

NoSQL : Graphes

- S'appuie sur théorie des graphes
- Notions de noeuds, relations et propriétés
- Une requête devient une traversée de graphe

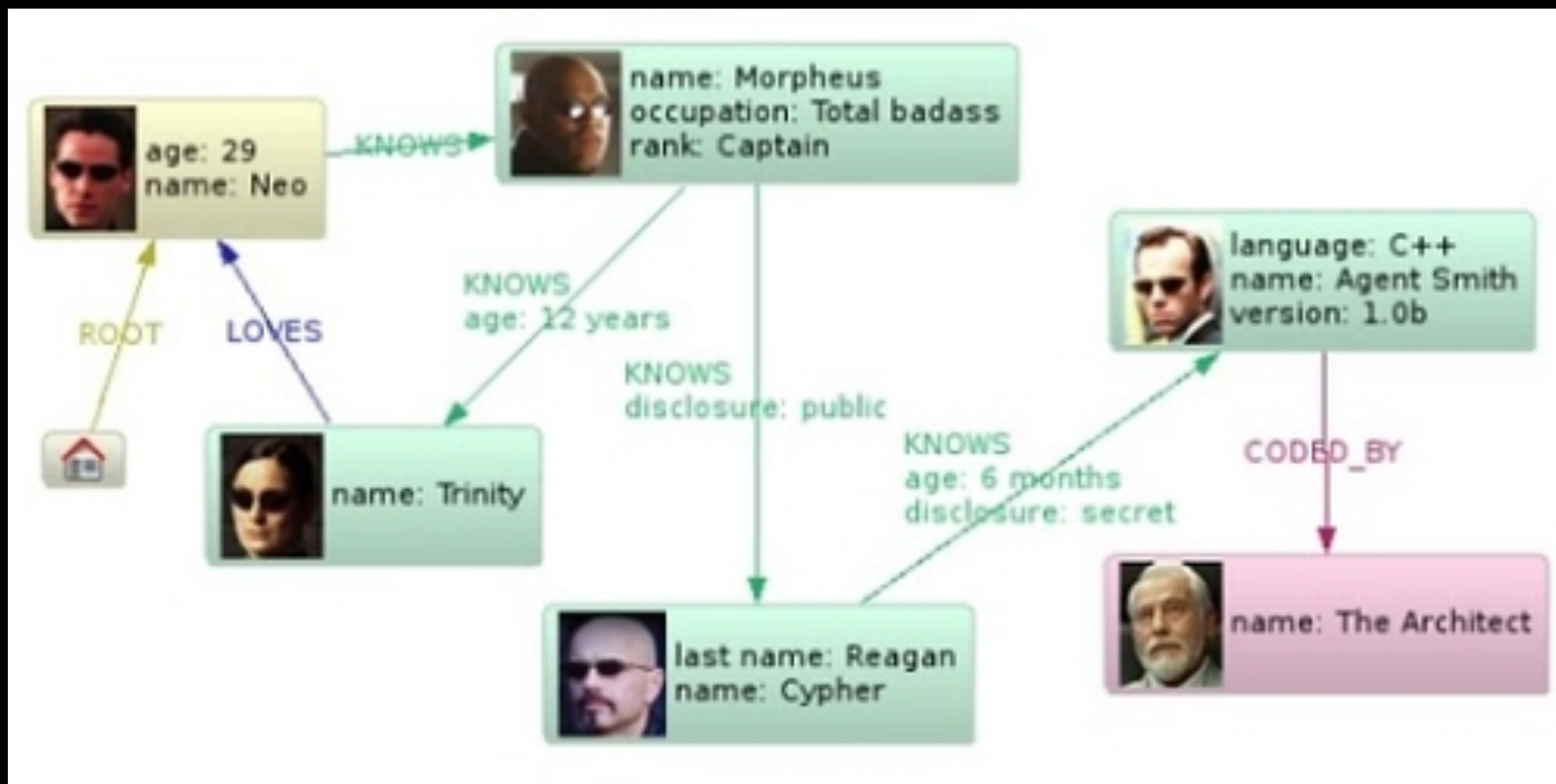
NoSQL : Graphes



Crédit : InfoQ

NoSQL : Graphes

Réseau social de Matrix



Crédit : InfoQ

NoSQL : Graphes

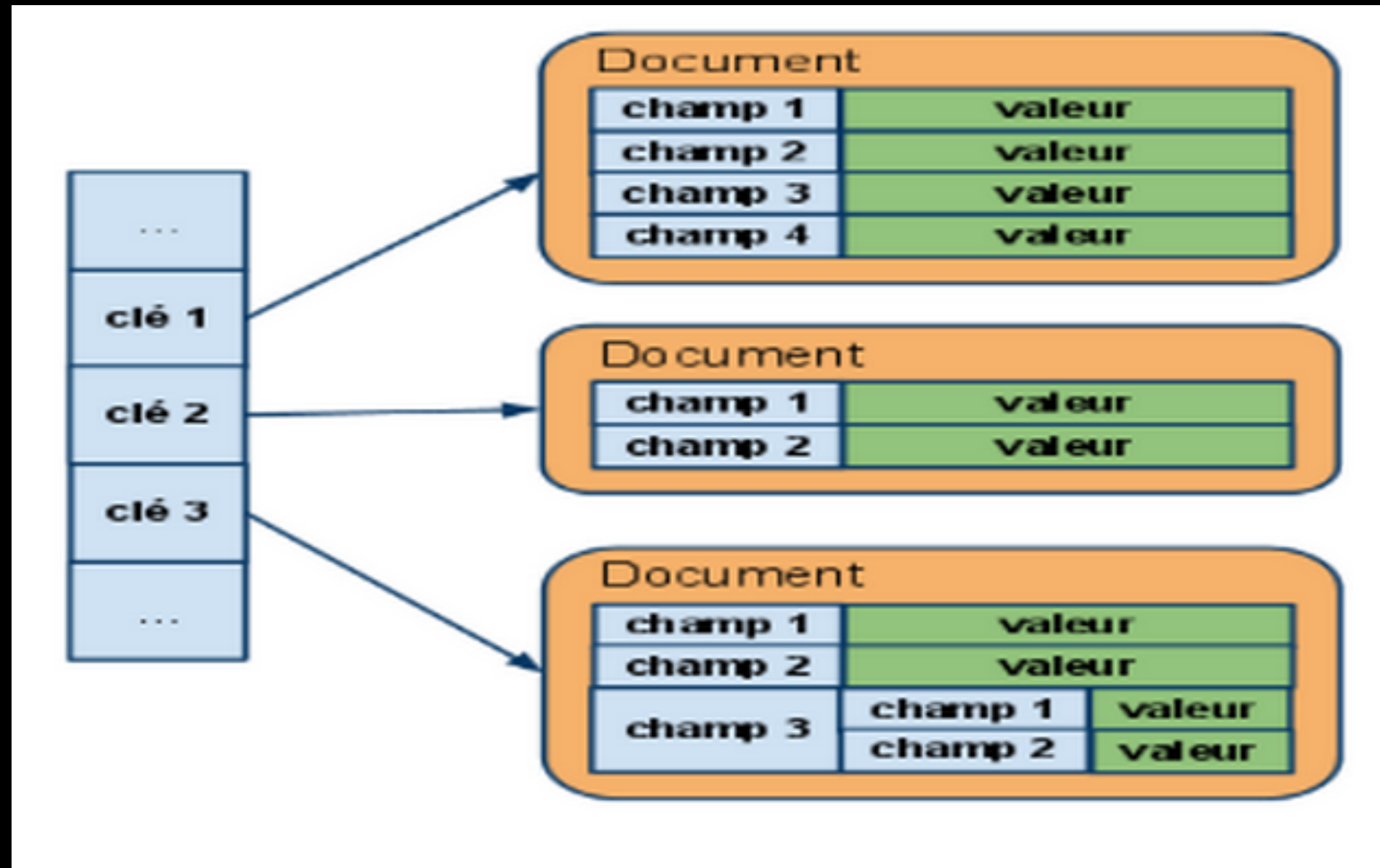


etc.

NoSQL : orienté document

- Proche de Clé-Valeur
- Clé-Document JSON ou XML
- Une Clé => un Ensemble structuré

NoSQL : orienté document



Crédit : Echeverria Portfolio

NoSQL : orienté document



etc.

One more thing....

One more thing

- Beaucoup de choix
- Que choisir ?
- Langage d'interrogation, système de stockage et modèle de données fortement liés

NoSQL : orientés multi-modèles

- « NoSQL de seconde génération »
- Répond aux problématiques précédentes :
 - Plusieurs modèles
 - Pour plusieurs applications
- Pour une même plateforme

NoSQL : orientés multi-modèles

- Explications :

Prenons Facebook...

Table Femme et Homme

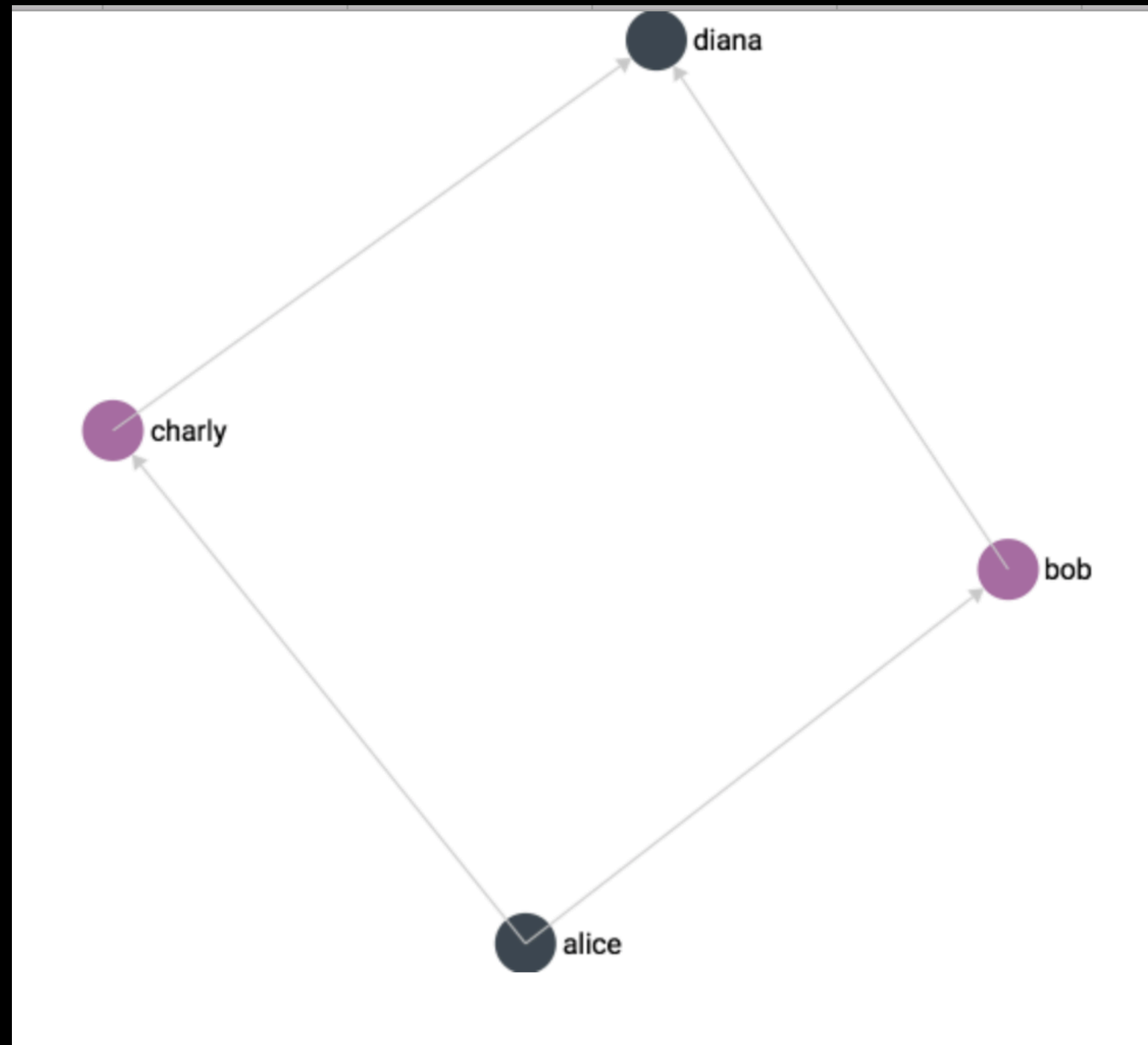
Nombreuses requêtes sur une femme ou un homme => Clé <=> Document

NoSQL : orientés multi-modèles

- Prenons à présent deux femmes et deux hommes :
 - Alice
 - Dianna
 - Bob
 - Charly
- Avec les relations suivantes :
 - Alice mariée à Bob
 - Charly marié à Dianna
 - Alice amie de Charly
 - Bob ami de Dianna

NoSQL : orientés multi-modèles

- Meilleure représentation : Graphe
- Ah oui mais j'ai déjà toutes les données dans mon orientée document...



Exemple d'ArangoDB

NoSQL : orientés multi-modèles

- Avantages :
 - Flexible
 - Moins coûteuse => BDD globale et plus locale
 - Cohérence des données => Système de stockage « universel » pour tous les modèles

NoSQL : orientés multi-modèles

- Désavantages :
 - Très très récent => Pas de feedback sur la longue durée
 - Peu de personnes formées actuellement dessus (occasion à saisir ?)

NoSQL : orientés multi-modèles



(racheté par Apple en Mars 2015)



etc.

JSON

- JavaScript Object Notation
- Prononcé « geille - zone » (comme Jason Statham)
- Format ouvert de structuration de document
- Utilise la syntaxe JavaScript

```
{  
  "tracks": [{  
    "name": "J'ai demandé à la lune",  
    "artist": {  
      "name": "Indochine"  
    }  
  }, {  
    "name": "Trouble",  
    "artist": {  
      "name": "ColdPlay"  
    }  
  }  
}]  
}
```



Crédit : Jason Statham

JSON

- Quelques avantages de JSON sur XML
 - Moins verbeux (pas de balises fermante)
 - Peut contenir des tableaux
 - Particulièrement pratique pour développement Web (syntaxe JavaScript et parsing immédiat)
- Utilisé indirectement dans MongoDB

JSON

- Quelques désavantages :
 - Peu de structure de données (pas d'ensemble par exemple)
 - Pas de xQuery ou XPath

MongoDB



- « huMONGOus DataBase » (BDD gigantesque)
- Développée par Dwight Merriman et Eliot Horowitz
- Écrit en C++, interaction en JS, APIs pour Java, PHP...
- Réplication: maître-esclave
- Passage à l'échelle: sharding automatique

MongoDB

- Orienté Document
- Principe :
 - Une clé -> Un Document
 - Document : structure ordonnée connue de la base
 - Interrogation par requêtes



MongoDB

- Idéal pour :
 - Requêtes dynamiques
 - Applications en lien avec données mais pas de relation entre elles autres que hiérarchiques

MongoDB : données

- Exemple de données appropriées :
 - Séries télévisées : séries > saisons > épisodes > acteurs
- Exemple de données non appropriées :
 - Séries télévisées : séries > saisons > épisodes > acteurs > séries de ces acteurs

Quel type de bdd à votre avis pour cet exemple ?

MongoDB : communication

- Communication
 - Interface en JS
- Exemple

```
> db.createCollection("test");  
> db.test.insert({title:"MongoDB",categories:["NoSQL","Database"]}); >  
var docs = db.test.find({categories:"NoSQL"});  
> docs.categories = ["NoSQL","Document Database"];  
> db.test.save(docs);  
> ...  
> db.test.find().sort({title: 1}).skip(20).limit(10);  
> ...
```

MongoDB : format des documents



Crédit : Un Bison

BSON: Binary JSON

- Types de données: string, integer, double, boolean, date, byte array, objets BSON et arrays, null
- Efficace en consommation mémoire et vitesse de traitement
- BSON est un sur-ensemble de JSON

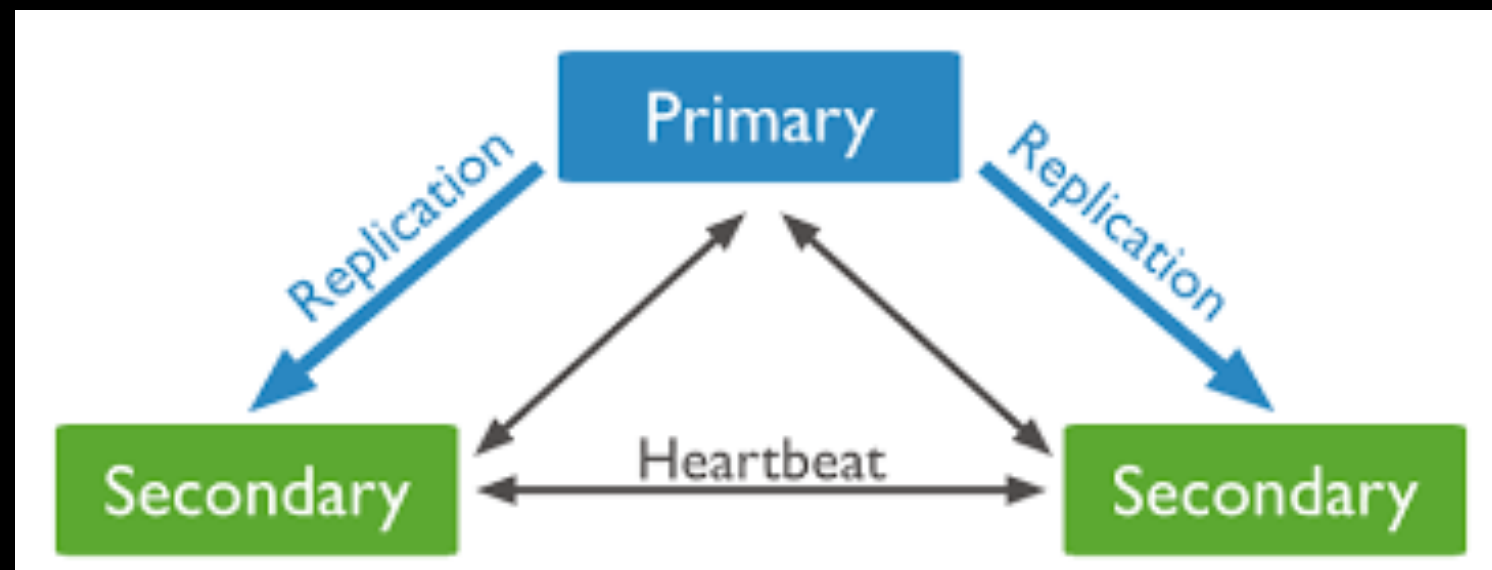
MongoDB : index

Index

- croissant
- décroissant
- multiples
- unique

MongoDB : réplication

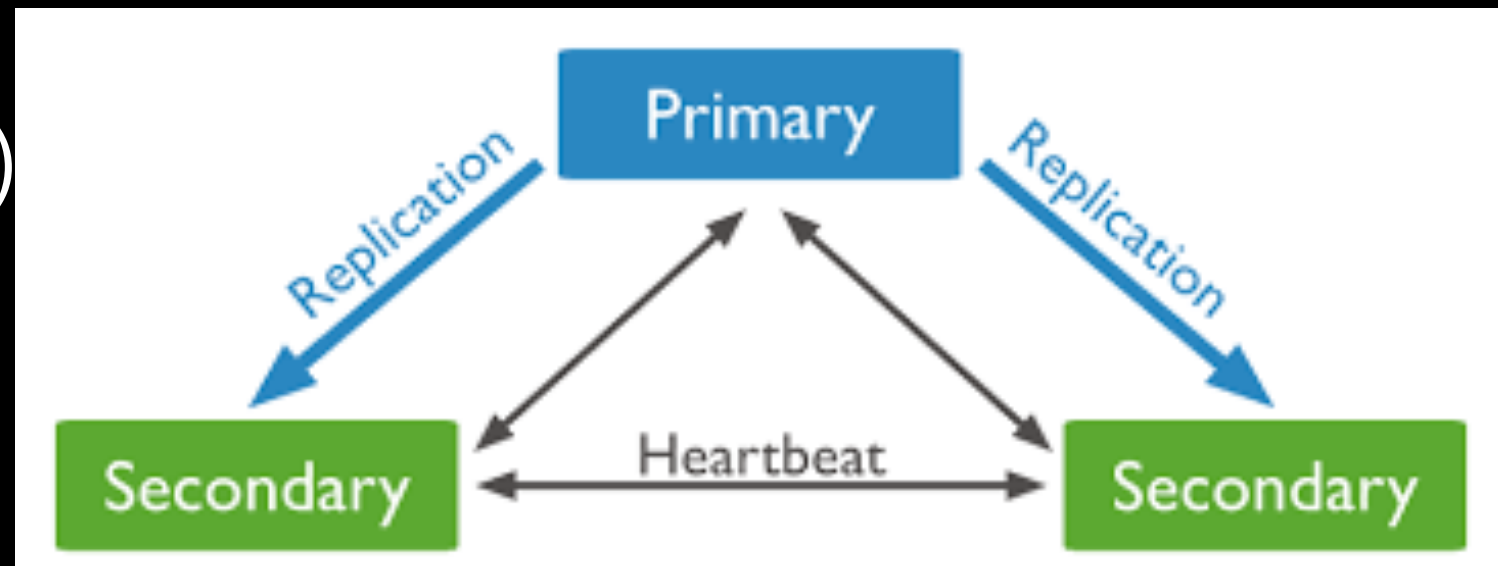
- Réplication Maître-esclave asynchrone
- Déployée sur au moins 2 serveurs



Crédit : Documentation MongoDB

MongoDB : réplication

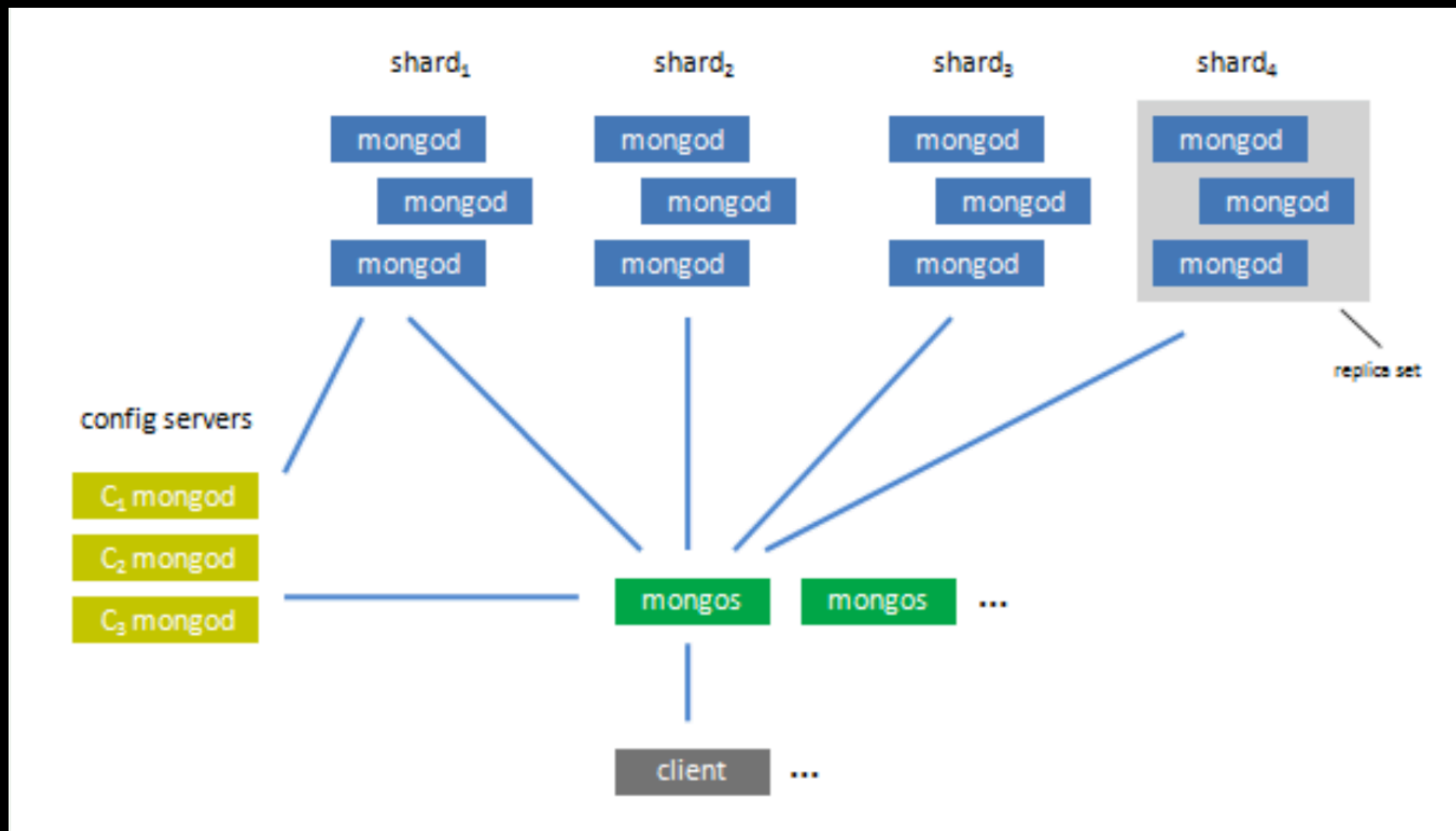
- Un seul nœud (master) est responsable des écritures
- Lectures sur ce nœud pour consistance forte, ou sur répliqués pour consistance éventuelle



Crédit : Documentation MongoDB

MongoDB : Échelle horizontale

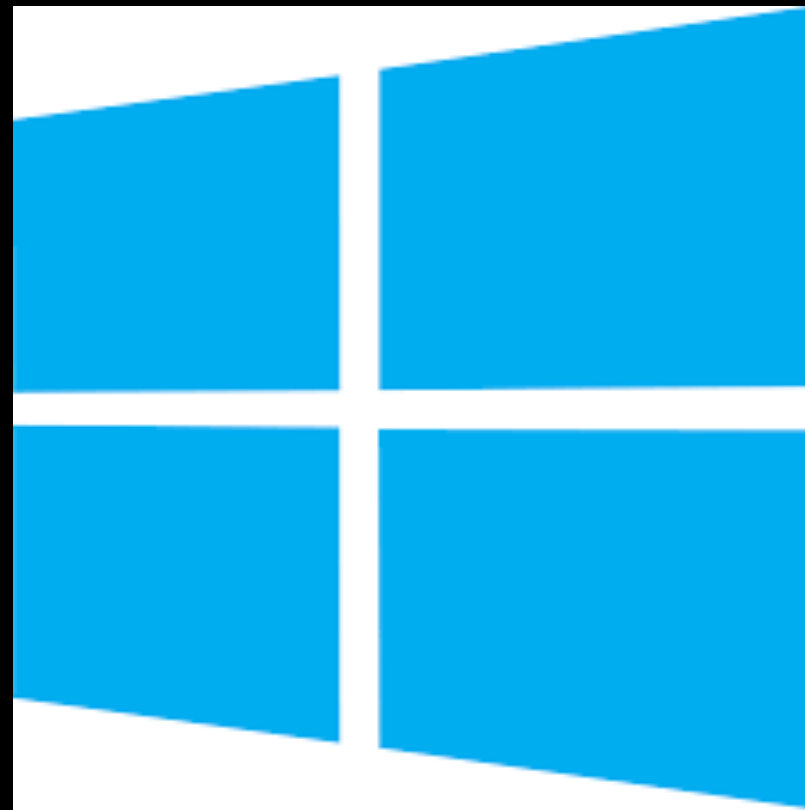
Auto-Sharding (redimensionnement automatique)



Crédit : Loïc Guillot

MongoDB : Plateforme

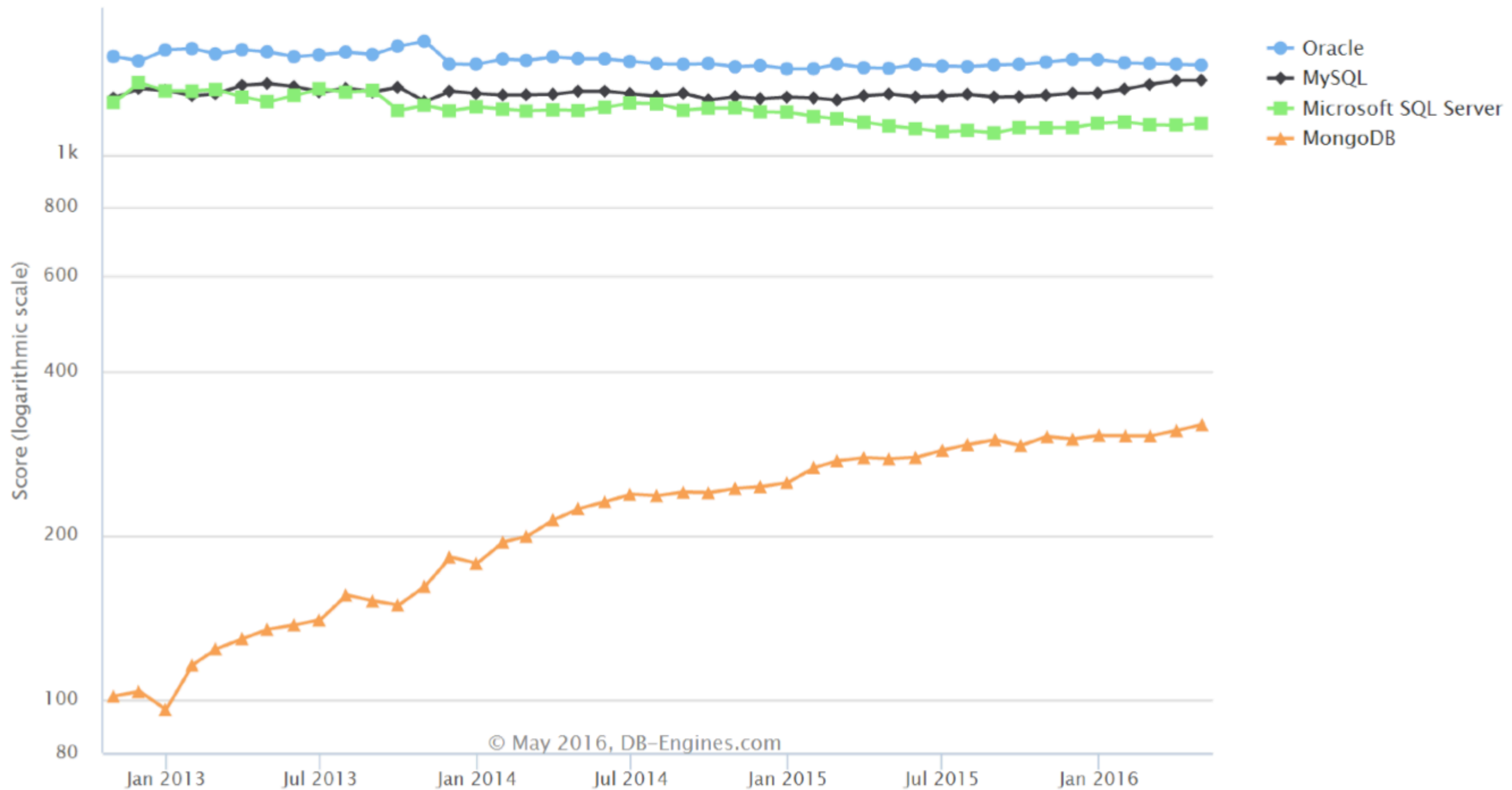
Cross-platform



Pourquoi choisir MongoDB ?

- Sharding automatique
- Système de fichier : GridFS (sharding sur fichier > 4 Mo)
- Requête aussi riche qu'en SQL
- Dynamique

MongoDB



Crédit : DB-Engines

MongoDB ou CouchDB ?

	MongoDB	CouchDB
Scalabilité	++	+++ (Erlang)
Monitoring	+	++
Requêtage	+++	+ (vues)
Performance	+	++

MongoDB : Licences

- 2 types :
 - OpenSource (AGPL) + drivers sous Apache
 - Support payant

MongoDB : quelques utilisateurs



Crédit : Site Officiel MongoDB

MongoDB : Partenariat



Stockage des données dans HDFS et Map/Reduce pour les traitements

MongoDB : Conclusion

Intérêt des bases orientés documents :

- Facile à utiliser
 - Manipulation de collections d'objets
 - Accès en REST

MongoDB : Conclusion

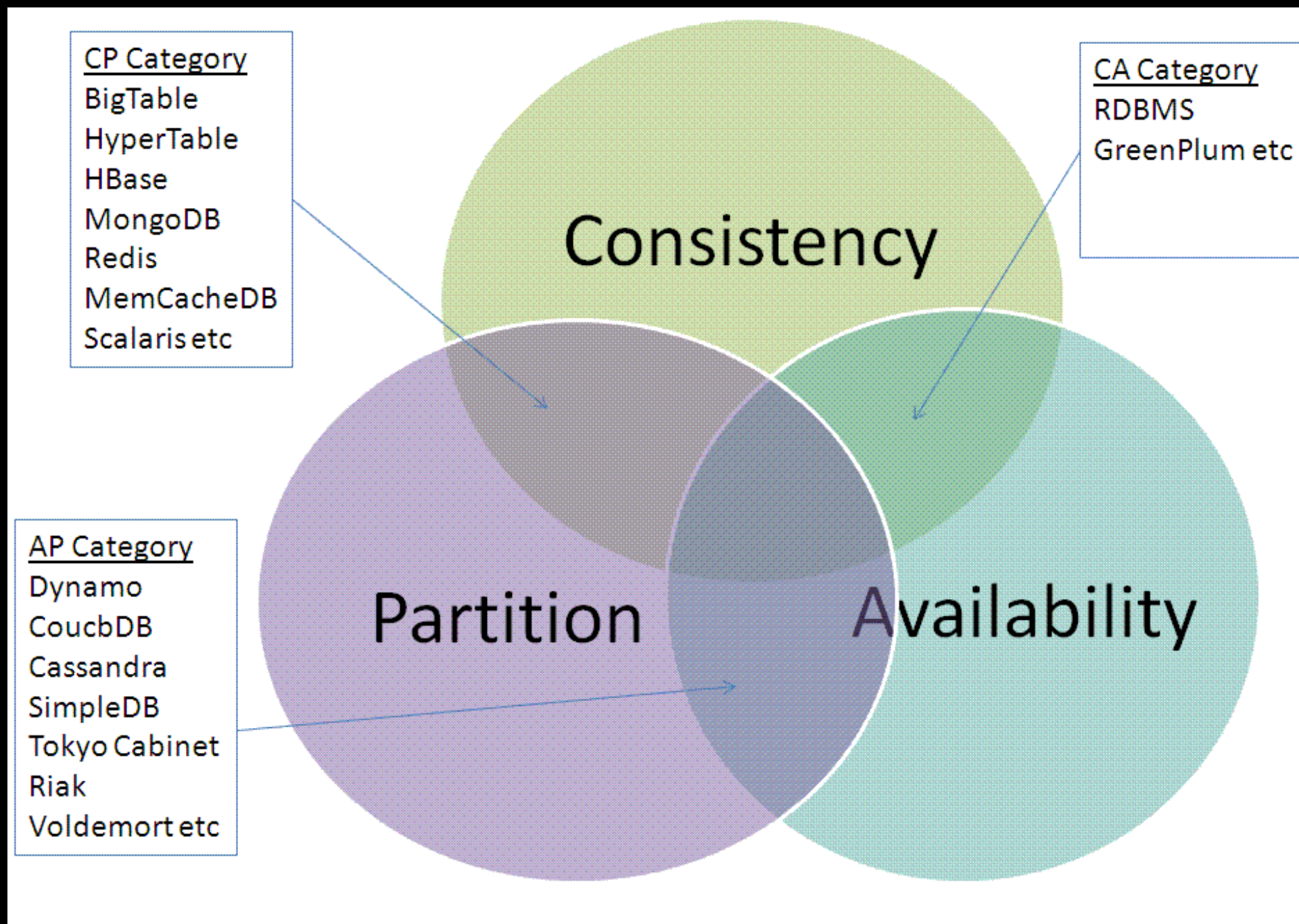
- Rapide
 - Pas de jointure pour retrouver les données dépendantes
- Flexible
 - Évolutivité du schéma

NoSQL : Conclusion

	Performances	Passage échelle horizontale	Variété données	Complexité opération
Clé-Valeur	+++	+++	+++	--
Colonne	+++	+++	+	-
Document	+++	+	+++	-
Graphe	+	+	+++	+++
SGBDR	+	+	-	+

NoSQL : Conclusion

- Différentes technologies \Leftrightarrow différents besoins



Crédit : Deimos

NoSQL : Conclusion

- Utilisés par des grands noms (Google, eBay, Twitter, etc.)
- Permet de stocker beaucoup beaucoup de données pour les traiter
- Constante évolution

Sources

- <https://www.mongodb.com/>
- <http://www.lemagit.fr/conseil/Lessentiel-sur-MongoDB>
- <https://en.wikipedia.org/wiki/MongoDB>
- <https://openclassrooms.com/courses/guide-de-demarrage-pour-utiliser-mongodb>
- <http://dev.af83.com>
- <https://docs.mongodb.com/manual/core/gridfs/>
- <http://blog.xebia.fr/2010/05/04/nosql-europe-bases-de-donnees-orientees-colonnes-et-cassandra/>
- <http://blog.neoxia.com/nosql-5-minutes-pour-comprendre/>

Sources

- <https://www.arangodb.com>
- <http://fr.intersystems.com/blog/avantages-concurrentiels-base-de-donnees-multi-modeles>
- <http://blog.romainschlick.com/arangodb-une-base-nosql-multi-modele/>
- Et les crédits de toutes les images

Avez-vous des Questions ?

Alexandre GINGEMBRE
Mathieu MOUROT



TPs

- Dans un 1er temps, HelloWorld avec MongoDB
- Puis un TP avec Java-MongoDB

TP HelloWorld

- Pré-requis : Avoir MongoDB d'installé

TP Java-MongoDB

- Pré-requis :
 - Avoir MongoDB d'installé
 - Avoir Java (JDK)
 - Avoir Maven
 - Avoir Git
 - Avoir PostMan