

MIF18 - Les SGBD Non-Relationnels

Fabien Duchateau

fabien.duchateau [at] univ-lyon1.fr

Université Claude Bernard Lyon 1

2013 - 2014



Transparents disponibles sur

<http://liris.cnrs.fr/~ecoquery/dokuwiki/doku.php?id=enseignement:bdav:start>

Introduction

Depuis les années 1970, dominance du modèle Relationnel

Avec l'émergence du Web et d'Internet, se pose le problème du passage à l'échelle (millions d'utilisateurices interagissant avec un système donné)

Réflexions pour passer d'un système large échelle vertical ("dopage" du serveur) à un système large échelle horizontal (ajout de machines)

Enfin, explosion du volume de données à stocker et traiter (phénomène "Big Data")

Plan du cours

Le « Big Data »

Généralités sur les BD NoRel

Modélisation d'une BD NoRel

Définition

« **Big Data** » : modélisation, stockage et traitement d'un ensemble de données très volumineuses, croissantes et hétérogènes, dont l'exploitation permet entre autre :

- ▶ Prise de décisions
- ▶ Découverte de nouvelles connaissances
- ▶ Possibilités de nouveaux « business models » (e.g., accès à un service contre des informations)

Causes :

- ▶ Faible coût du stockage
- ▶ Faible coût des processeurs
- ▶ Mise à disposition des données

Quelques applications du Big Data

Sciences :

- ▶ Création de cartes de navigation par Fontaine Maury, au 19^{eme} siècle, à partir de vieux journaux de bord (précurseur)
- ▶ Large Hadron Collider (LHC), un accélérateur de particules qui génère un flux de 40 millions de données par seconde
- ▶ Linked Open Data (e.g., DBpedia, Freebase)

Politique :

- ▶ Prédiction du résultat des élections États-uniennes 2012
- ▶ Transparence (Open Data)



http://en.wikipedia.org/wiki/Big_data#Big_science

<http://linkeddata.org>

http://fr.wikipedia.org/wiki/Open_data

Quelques applications du Big Data (2)

Industrie, secteur public, santé, etc. :

- ▶ Amazon gère des millions d'opérations par jour
- ▶ Programmes de surveillance (e.g., Prism)
- ▶ Décodage du génome humain
- ▶ Prédiction des analystes de Google quelques semaines avant l'épidémie de grippe en 2009
- ▶ Découverte d'un effet secondaire dû à la prise de deux médicaments par analyse des requêtes des internautes (Yahoo)
- ▶ Confirmation de la censure par analyse de la fréquence de noms de personnes (Chagall en Allemagne, Trotsky en URSS)

Caractéristiques

Les « 3V », caractéristiques du Big Data :

- ▶ **Volume** (plusieurs zettaoctets générés par an sur le Web)
- ▶ **Vélocité** (fréquence de génération des données)
 - ▶ notion de flux (stream)
 - ▶ 7 Go par jour pour Twitter (2012)
 - ▶ 7000 To par seconde prévus pour le radiotélescope "Square Kilometre Array"
- ▶ **Variété** (hétérogénéité)
 - ▶ données brutes, structurées ou pas, etc.
 - ▶ images, texte, géo-démographiques, profils utilisateurices, etc.

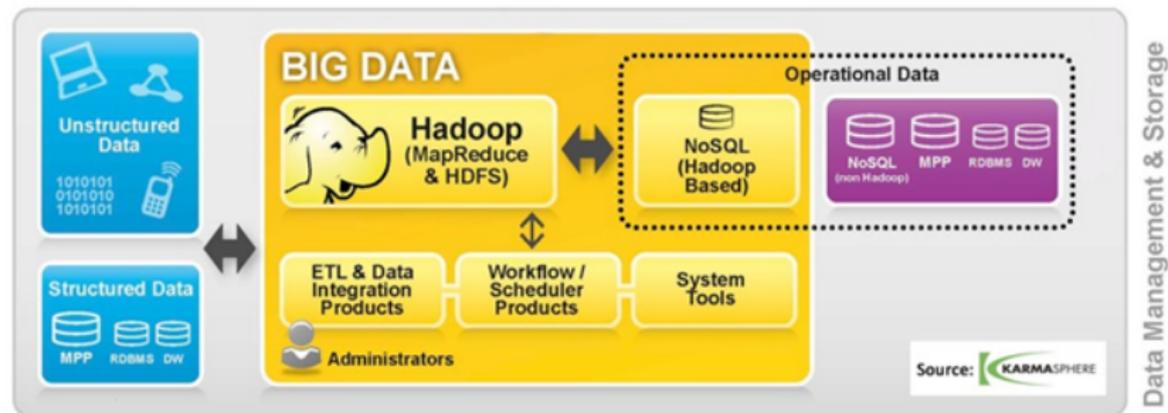
http://fr.wikipedia.org/wiki/Big_data

<http://fr.wikipedia.org/wiki/Zettaoctet>

Caractéristiques (2)

Philosophie : acquisition de données (structurées ou pas) puis organisation de ces données

Importance des outils statistiques pour l'analyse



Un exemple

Exemple d'une usine qui fabrique des chamallows :

- ▶ Problème : les ventes de chamallows baissent



Un exemple

Exemple d'une usine qui fabrique des chamallows :

- ▶ Problème : les ventes de chamallows baissent
- ▶ Données : les ventes historisées par client, région, etc.
- ▶ Information : l'achat de chamallows se fait principalement par des adultes



Un exemple

Exemple d'une usine qui fabrique des chamallows :

- ▶ Problème : les ventes de chamallows baissent
- ▶ Données : les ventes historisées par client, région, etc.
- ▶ Information : l'achat de chamallows se fait principalement par des adultes
- ▶ Connaissances :
 - ▶ les chamallows sont essentiellement mangés par des jeunes
 - ▶ pour les parents, les chamallows provoquent des caries



Un exemple

Exemple d'une usine qui fabrique des chamallows :

- ▶ Problème : les ventes de chamallows baissent
- ▶ Données : les ventes historisées par client, région, etc.
- ▶ Information : l'achat de chamallows se fait principalement par des adultes
- ▶ Connaissances :
 - ▶ les chamallows sont essentiellement mangés par des jeunes
 - ▶ pour les parents, les chamallows provoquent des caries
- ▶ Solution : démarcher les dentistes pour promouvoir les chamallows, etc.



Un exemple (2)

L'usine possède certaines données :

- ▶ Ventes historisées (e.g., entrepôt de données)
- ▶ Informations sur les acheteurs (e.g., analyse des ventes)

Mais d'autres données se trouvent ailleurs :

- ▶ Connaissances (enquêtes clients, flux de données, profils, etc.)

⇒ **Besoin d'autres sources de données, et d'interroger intelligemment ces données**



En résumé

Phénomène du « Big Data » : volume, vélocité, variété

De nouveaux besoins émergent :

- ▶ Au niveau du stockage
- ▶ Au niveau des modèles de données
- ▶ Au niveau du traitement



Plan du cours

Le « Big Data »

Généralités sur les BD NoRel

Modélisation d'une BD NoRel

Contexte

Contexte du Big Data :

- Demande émergente pour des SGBD distribués
 - à forte disponibilité
 - résistants au morcellement

Contexte des applications Web :

- Schémas dynamiques (nombre d'attributs extensible, pas de valeurs nulles)
- Nombre important de lectures/écritures
- Relations complexes

Contexte

Contexte du Big Data :

- Demande émergente pour des SGBD distribués
 - à forte disponibilité
 - résistants au morcellement

Contexte des applications Web :

- Schémas dynamiques (nombre d'attributs extensible, pas de valeurs nulles)
- Nombre important de lectures/écritures
- Relations complexes

Les SGBD Relationnels sont moins adaptés pour le Big Data et le Web ⇒ mouvement NoSQL, NotOnlySQL, NoRel, NewSQL, ...

Théorème CAP

Le théorème de Brewer ou **théorème CAP** = un système distribué ne peut garantir en même temps les trois propriétés suivantes :

- ▶ Cohérence (consistency) : tous les noeuds du système voient la même information au même moment
- ▶ Disponibilité (*availability*) : toute requête reçoit une réponse
- ▶ Résistance au morcellement (*partition tolerance*) : fonctionnement autonome en cas de morcellement du réseau (sauf panne totale)

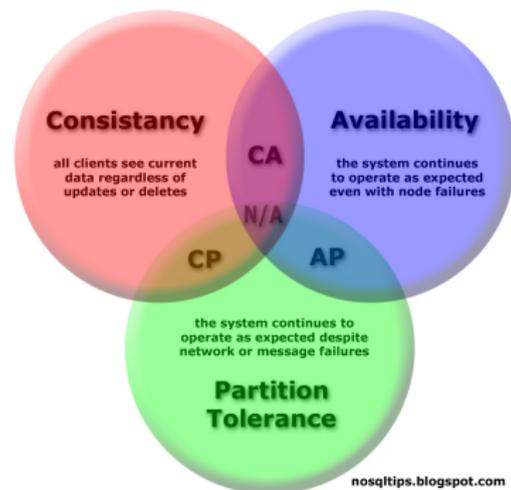
<http://ksat.me/a-plain-english-introduction-to-cap-theorem/>

<http://www.julianbrowne.com/article/viewer/brewers-cap-theorem>

Caractéristiques des BD NoRel

Caractéristiques :

- ▶ Garantie de deux propriétés parmi cohérence, disponibilité et résistance au morcellement
- ▶ Performance (scalabilité horizontale)
- ▶ Pas de schéma de table fixé
- ▶ Eviter les coûteuses jointures (données dénormalisées)
- ▶ Pas de transactions
- ▶ Pas de fonctionnalité de requêtage complexe (SQL)

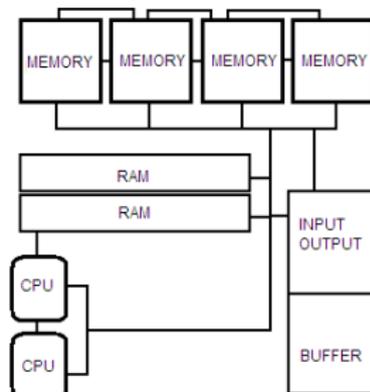


Caractéristiques des BD NoRel (2)

BD NoRel \approx entrepôt clé-valeur fortement optimisé

Les BD NoRel sont dites BASE, pour :

- ▶ Basically Available (haute disponibilité)
- ▶ Soft-state (changement de l'état du système, même sans mise à jour)
- ▶ Eventually consistent (cohérence sur le long terme)

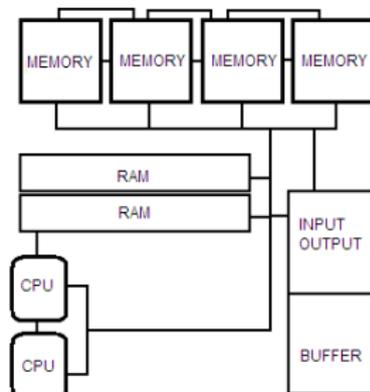


Caractéristiques des BD NoRel (2)

BD NoRel \approx entrepôt clé-valeur fortement optimisé

Les BD NoRel sont dites BASE, pour :

- ▶ Basically Available (haute disponibilité)
- ▶ Soft-state (changement de l'état du système, même sans mise à jour)
- ▶ Eventually consistent (cohérence sur le long terme)



Comment obtenir cette cohérence sur le long terme ?

Cohérence sur le long terme

Dans la philosophie NoRel, on accède aux données par des clés

Cohérence d'une clé (distribuée) : toutes les lectures pour cette clé ont la même valeur

La cohérence est un problème lors d'écritures simultanées de données sur des serveurs distribués (voire sur des centres de données distribués)

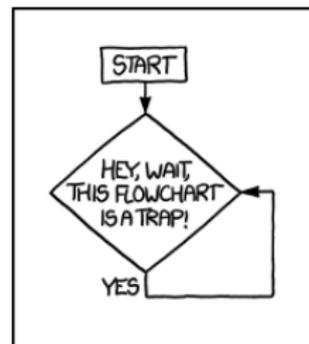
- ▶ Solution en Relationnel avec les transactions distribuées
- ▶ Trop lente et dépendance à la propriété de disponibilité

⇒ Propositions de technique pour la cohérence à long terme

Cohérence sur le long terme (2)

Techniques pour atteindre la cohérence à long terme :

- ▶ **Two-Phase Commit** : protocole qui coordonne les processus impliqués dans une transaction atomique distribuée à commiter ou annuler (lent et non tolérant aux pannes)
- ▶ **Paxos-style consensus** : protocole qui trouve une valeur au consensus parmi les processus participants



http://en.wikipedia.org/wiki/Two-phase_commit_protocol

http://en.wikipedia.org/wiki/Paxos_algorithm

Cohérence sur le long terme (3)

Techniques pour atteindre la cohérence à long terme :

- ▶ **Read-repair** : écriture de toutes les versions incohérentes, et lors d'une prochaine lecture, le conflit sera détecté et résolu (souvent en écrivant sur un certain nombre de répliques)

Exemple avec Dynamo (Amazon) : dans un panier, les ajouts de produits sont cruciaux, mais les suppressions de produits le sont moins car le client corrigera l'erreur. Dans ce cas, l'union du contenu des deux paniers en conflit permet de ne pas perdre d'ajout de produits. Utilisation des horloges vectorielles pour limiter l'incohérence des paniers au niveau des suppressions de produits

Philosophie "When in doubt, take customer's order!"

<http://the-paper-trail.org/blog/2008/08/26/>

http://en.wikipedia.org/wiki/Vector_clock

Catégories des BD NoRel

Domaine en pleine évolution !

Proposition de classements selon :

- ▶ Le modèle de données
- ▶ Les caractéristiques non fonctionnelles
- ▶ Les propriétés du théorème de CAP

Dans la suite, classement selon le modèle de données

Steven Yen, *NoSQL is a horseless carriage*, 2009, <http://dl.getdropbox.com/u/2075876/nosql-steve-yen.pdf>

<http://fr.slideshare.net/bscofield/nosql-codemash-2010>

Le modèle Relationnel

Un modèle que vous connaissez bien...

- Relations, attributs, tuples, etc.
- Propriétés de cohérence et de disponibilité
- SGBD : PostgreSQL, Oracle, MySQL, etc.

Exemple de tables Relationnelles

Table ECRIVAIN

id	nom	pays	dateNaiss
2	GRR Martin	USA	20/09/48

Table LIVRE

id	titre	prix	datePubli	ecrivain
1	Le trône de fer	15	01/08/96	2

Entrepôt clé-valeur

Entrepôt clé-valeur = ce modèle est aussi appelé « key-value store » ou tableau associatif

- ▶ La clé est un identifiant unique
- ▶ La valeur peut être structurée ou pas

Implémentation minimale :

1. valeur = **get**(clé)
2. **insert**(clé, valeur)
3. **delete**(clé)

Les implémentations proposent souvent des fonctionnalités ou propriétés supplémentaires

Entrepôt clé-valeur (2)

SGBD type entrepôt clé-valeur :

- ▶ Serveur standalone (Redis)
- ▶ Distribué (Dynamo, Riak, Voldemort)
- ▶ Uniquement en mémoire (Memcached)
- ▶ ...

Performances (++) , passage à l'échelle (++) , flexibilité (++) , prise en main (++) , nombre de fonctionnalités (~/-)

<http://redis.io/>

<http://www.allthingsdistributed.com/files/amazon-dynamo-sosp2007.pdf>

<http://basho.com/riak/>

<http://www.project-voldemort.com/>

<http://www.memcached.org/>

Entrepôt clé-valeur (3)

Exemple d'un entrepôt clé-valeur

```
"nom-ecrivain2" : "GRR Martin"  
"pays-ecrivain2" : "USA"  
"dateNaiss-ecrivain2" : 20/09/48  
"titre-livre1-ecrivain2" : "Le trône de fer"  
"prix-livre1-ecrivain2" : 15  
"datePubli-livre1-ecrivain2" : "01/08/96"
```

BD orientée colonnes

BD orientée colonnes = organisation des données en colonnes

- ▶ Une colonne stocke le nom de la colonne et la valeur associée (et un timestamp)
- ▶ Une supercolonne stocke des colonnes (\approx ligne en Relationnel)
- ▶ Une famille de colonnes stocke des colonnes ou supercolonnes

Avantages :

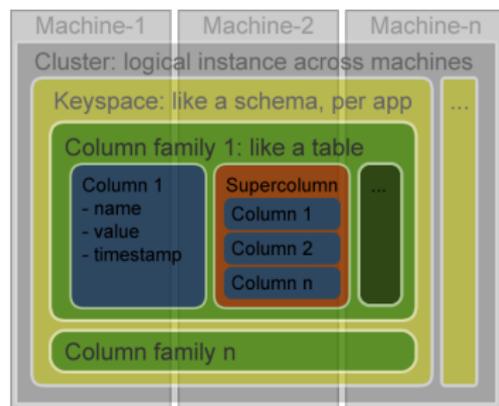
- ▶ Schéma dynamique (ajout de colonne)
- ▶ Pas de stockage de valeurs nulles
- ▶ Possibilité d'avoir des "super colonnes" pour stocker des listes de listes (e.g., Cassandra)

Ne pas confondre avec des BD Relationnelles orientées colonnes, qui sérialisent les données par colonne (e.g., MonetDB, Vertica)

BD orientée colonnes (2)

SGBD orienté colonnes :

- ▶ BigTable (propriété de Google)
- ▶ Cassandra (implémentation libre de BigTable)
- ▶ HBase, Hypertable (basé sur BigTable et Hadoop DFS)
- ▶ ...



Performances (++), passage à l'échelle (++), flexibilité (+), prise en main (+), nombre de fonctionnalités (-)

<http://cassandra.apache.org/>
<http://cassandra-php.blogspot.fr/>
<http://hbase.apache.org/>
<http://hypertable.com/>
<http://en.wikipedia.org/wiki/Hadoop>

BD orientée colonnes (3)

Exemple d'une BD orientée colonnes

La première ligne représente la famille de colonne *écrivain*

La seconde ligne représente la famille de colonne *livre*

1	nom :	pays :	dateNaiss :	
	GRR Martin	USA	20/09/48	
2	titre :	prix :	datePubli :	auteur :
	Le trône de fer	15	01/08/96	1

BD orientée graphes

BD orientée graphes = éléments interconnectés avec un nombre indéterminé de relations entre eux

- ▶ Noeuds pour représenter des entités
- ▶ Arc étiqueté et directionnel entre deux noeuds
- ▶ Propriété sur un noeud ou un arc

Avantages :

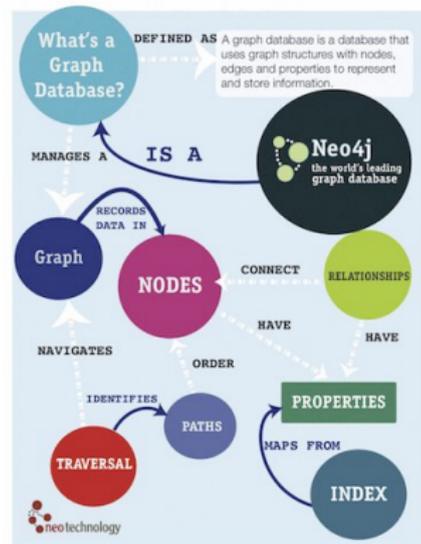
- ▶ Emergence de "patterns" par analyse des noeuds
- ▶ Facilité d'évolution du schéma
- ▶ Bonnes performances pour des requêtes type graphe (e.g., plus court chemin)
- ▶ Permet de représenter les 3 autres modèles

BD orientée graphes (2)

SGBD orienté graphes :

- ▶ Neo4J (ACID, avec transactions)
- ▶ Allegro Graph (triplestore, raisonnement Prolog)
- ▶ Virtuoso (triplestore et SGBDR)
- ▶ ...

Performances (~), passage à l'échelle (~), flexibilité (++), prise en main (-), fonctionnalités (théorie des graphes)



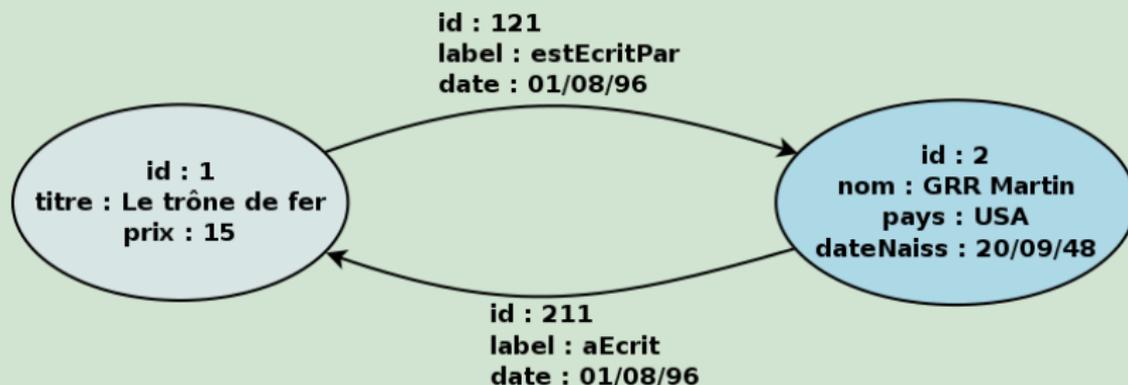
<http://www.neo4j.org/>

<http://www.franz.com/agraph/allegrograph/>

<http://virtuoso.openlinksw.com/>

BD orientée graphes (3)

Exemple d'une BD orientée graphes



BD orientée documents

BD orientée documents = collection de documents (clé-document)

- ▶ Notion abstraite de « document »
- ▶ Chaque document a des champs et une clé
- ▶ Deux instances de document peuvent avoir des champs différents
- ▶ Organisation des documents selon des tags, métadonnées, collections

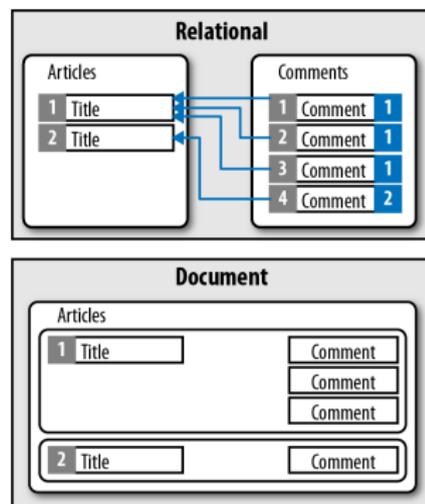
Avantages :

- ▶ Recherche de documents basée sur leur contenu
- ▶ Facilité d'évolution du schéma

BD orientée documents (2)

SGBD orienté documents :

- ▶ CouchDB (stockage JSON, requêtes en Javascript via Map Reduce)
- ▶ MongoDB (documents "à la JSON", requêtes en Javascript)
- ▶ Couchbase (documents JSON)
- ▶ ...



Performances (++), passage à l'échelle (~/++), flexibilité (++), prise en main (+), nombre de fonctionnalités (~/-)

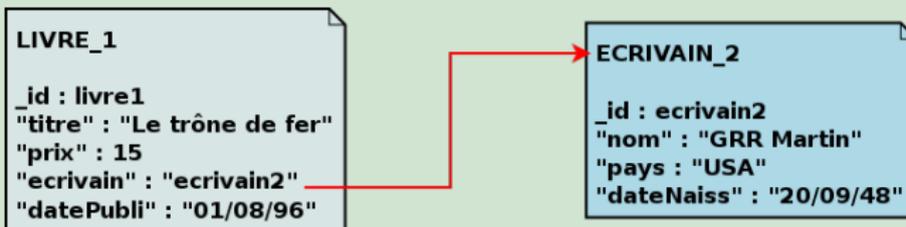
<http://couchdb.apache.org/>

<http://www.mongodb.org/>

<http://www.couchbase.com/>

BD orientée documents (3)

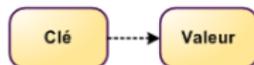
Exemple d'une BD orientée documents



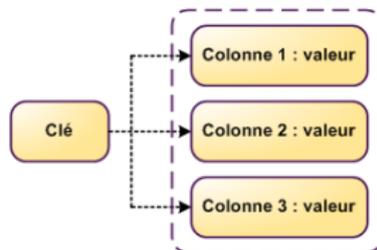
OU



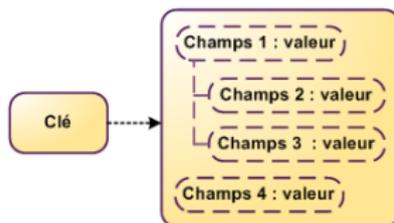
Synthèse des quatre catégories de BD No-Rel



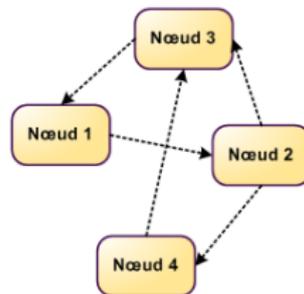
BDD Clé-Valeur



BDD Orientée colonnes



BDD Orientée document



BDD Orientée graphe

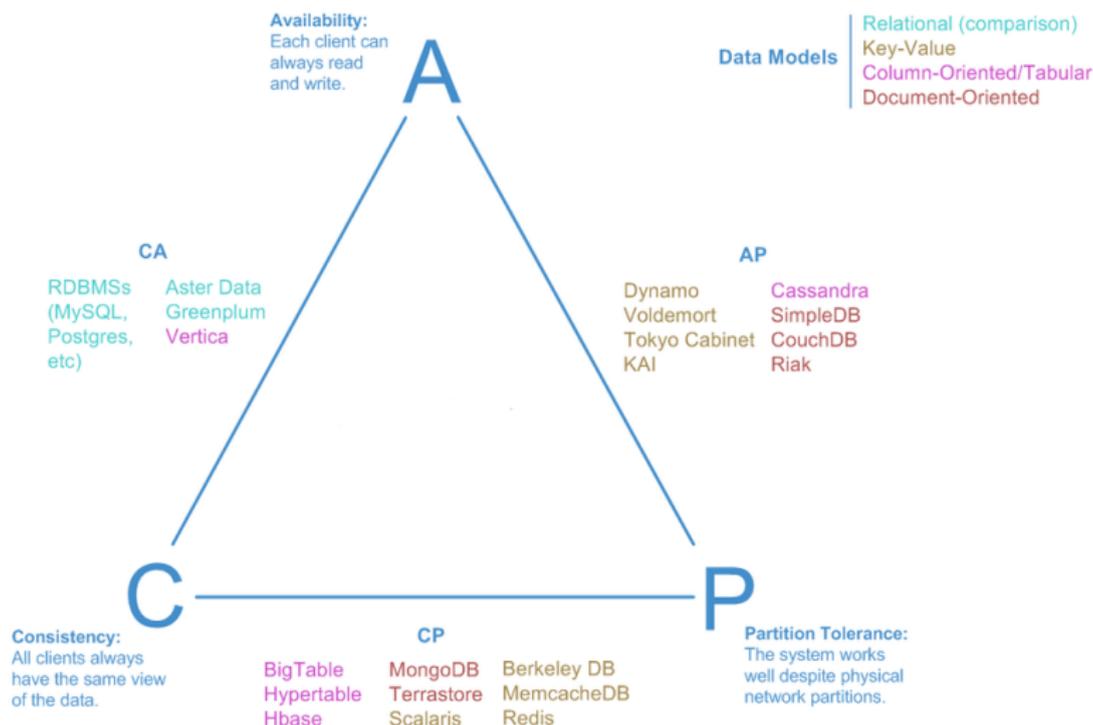
D'autres catégories de BD NoRel (2)

Émergence ou regain d'intérêt pour d'autres modèles :

- ▶ BD multi-modèles : OrientDB, AlchemyDB
- ▶ BD multi-dimensionnelles (matrices creuses) : Globals, GT.M
- ▶ BD multi-valeurs (non respect de la première forme normale, i.e., "pas d'attributs multi-valués") : Reality, OpenQM
- ▶ BD basée sur les événements : Event Store
- ▶ ...

Sont aussi considérés NoSQL des SGBD gérant du RDF ou des triplets (triplestores)

En résumé



Plan du cours

Le « Big Data »

Généralités sur les BD NoRel

Modélisation d'une BD NoRel

Conception d'une BD NoRel

Comment fait-on pour concevoir une BD NoRel au niveau physique ?

- ▶ Pas de tables, seulement des paires de clés-valeurs
- ▶ Tous les accès se font avec une clé !
- ▶ On peut ajouter un attribut/colonne à tout moment
- ▶ Une "valeur" peut être un objet complexe (liste, document, ensemble de valeurs, etc.)

Réflexion nécessaire pour identifier et stocker les concepts principaux d'une application, et les moyens d'y accéder

Exemple de conception d'une BD NoRel

Exemple de conception d'une BD *gestion de livres*

On souhaite une application Web qui permet à des utilisatrices d'évaluer et de commenter des livres qu'ils ou elles ont lus.

Utilisation du SGBD Redis pour l'exemple :

- ▶ Type entrepôt clés-valeurs
- ▶ Support d'objets (e.g., listes, ensembles, ensembles triés)
- ▶ Support de fonctions supplémentaires (e.g., union de deux ensembles, test de l'existence d'une clé)

Redis Data Examples

Keys	Values	
page:index.html	<html><head>[...]	← String
login_count	7464	
users_logged_in_today	{ 1, 2, 3, 4, 5 }	← Set
latest_post_ids	[201, 204, 209, ...]	← List
user:123:session	time => 10927353 username => joe	← Hash
users_and_scores	joe ~ 1.3483 bert ~ 93.4 fred ~ 283.22 chris ~ 23774.17	← Sorted (scored) Set

<http://redis.io/>

Exemple de conception d'une BD NoRel (2)

Concept **utilisateurice** :

- ▶ Quel identifiant ?

Le login est trop long comme clé, un identifiant numérique sert de clé

- ▶ Comment récupérer un identifiant utilisateurice ?

Une variable globale stocke la dernière valeur affectée pour l'insertion d'une utilisateurice (design pattern)

- ▶ Comment nommer les attributs ?

Le nom d'un attribut va inclure l'identifiant du concept

Insertion de l'utilisateurice *rat2biblio* identifiée avec *gruyere*

```
INCR global:nextUserId // retourne 1000
```

```
SET uid:1000:username rat2biblio
```

```
SET uid:1000:password gruyere
```

Exemple de conception d'une BD NoRel (3)

Concept **utilisateurice** :

- ▶ Problème : on n'accède à l'utilisateurice qu'avec sa clé !
Création d'une autre colonne qui retourne l'identifiant à partir du nom (impossible de retourner une clé pour une valeur donnée!)
- ▶ Comment assurer une authentification robuste ?
Ajout d'une longue chaîne de caractères aléatoire comme valeur d'un *auth cookie*

Gestion de la connexion pour l'utilisateurice *rat2biblio*

```
SET username:rat2biblio:uid 1000
```

```
SET uid:1000:auth abcde12345
```

```
SET auth:abcde12345:uid 1000
```

Exemple de conception d'une BD NoRel (4)

Concept **livre** (évalués par l'utilisatrice) :

- ▶ Besoin d'une variable globale pour le prochain identifiant d'un livre
- ▶ Construction des clés avec l'identifiant du livre et le nom de l'attribut
- ▶ Si besoin d'ajouter des colonnes (e.g., nom de l'auteur), on peut le faire plus tard !

Création d'un nouveau livre

```
INCR global:nextBookId // retourne 2
SET book:2:title "Le trône de fer"
SET book:2:prix 15
```

Exemple de conception d'une BD NoRel (5)

Concept **commentaire** (évaluation d'un livre) :

- Comment créer un commentaire pour un livre ?

Création d'un nouveau commentaire

Exemple de conception d'une BD NoRel (5)

Concept **commentaire** (évaluation d'un livre) :

- Comment créer un commentaire pour un livre ?

Création d'un nouveau commentaire

```
INCR global:nextReviewId // retourne 123
SET review:123:book 2
SET review:123:comment "Lecture conseillée"
SET review:123:uid 1000
```

Exemple de conception d'une BD NoRel (5)

Concept **commentaire** (évaluation d'un livre) :

- ▶ Comment créer un commentaire pour un livre ?
- ▶ Que faut-il ajouter pour retrouver tous les commentaires d'un livre ?

Création d'un nouveau commentaire

```
INCR global:nextReviewId // retourne 123
SET review:123:book 2
SET review:123:comment "Lecture conseillée"
SET review:123:uid 1000
```

Exemple de conception d'une BD NoRel (5)

Concept **commentaire** (évaluation d'un livre) :

- ▶ Comment créer un commentaire pour un livre ?
- ▶ Que faut-il ajouter pour retrouver tous les commentaires d'un livre ?

Création d'un nouveau commentaire

```
INCR global:nextReviewId // retourne 123
SET review:123:book 2
SET review:123:comment "Lecture conseillée"
SET review:123:uid 1000
SADD book:2:reviews 123 // reviews est un ensemble
```

Exemple de conception d'une BD NoRel (5)

Concept **commentaire** (évaluation d'un livre) :

- ▶ Comment créer un commentaire pour un livre ?
- ▶ Que faut-il ajouter pour retrouver tous les commentaires d'un livre ?
- ▶ Que faut-il ajouter pour retrouver tous les commentaires d'un-e utilisateurice ?

Création d'un nouveau commentaire

```
INCR global:nextReviewId // retourne 123
SET review:123:book 2
SET review:123:comment "Lecture conseillée"
SET review:123:uid 1000
SADD book:2:reviews 123 // reviews est un ensemble
```

Exemple de conception d'une BD NoRel (5)

Concept **commentaire** (évaluation d'un livre) :

- ▶ Comment créer un commentaire pour un livre ?
- ▶ Que faut-il ajouter pour retrouver tous les commentaires d'un livre ?
- ▶ Que faut-il ajouter pour retrouver tous les commentaires d'un-e utilisateurice ?

Création d'un nouveau commentaire

```
INCR global:nextReviewId // retourne 123
SET review:123:book 2
SET review:123:comment "Lecture conseillée"
SET review:123:uid 1000
SADD book:2:reviews 123 // reviews est un ensemble
SADD uid:1000:reviews 123 // reviews est un ensemble
```

Exemple de conception d'une BD NoRel (6)

Problèmes de la modélisation précédente, notamment pour le passage à l'échelle horizontale ?

Exemple de conception d'une BD NoRel (6)

Problèmes de la modélisation précédente, notamment pour le passage à l'échelle horizontale ?

- ▶ Réplication des données
- ▶ Incrémentation des variables globales

Exemple de conception d'une BD NoRel (6)

Problèmes de la modélisation précédente, notamment pour le passage à l'échelle horizontale ?

- ▶ Réplication des données
- ▶ Incrémentation des variables globales

Solutions possibles :

- ▶ Pour répliquer, on peut *hasher* la clé (e.g., modulo de la clé par le nombre de serveurs) et répartir ensuite les données sur les serveurs (cf cours MIF37)
- ▶ Pour l'incrémentation des variables globales :
 - ▶ avoir un serveur uniquement dédié aux incréments
 - ▶ générer comme clé une chaîne de caractères très longue (e.g., MD5, SHA-256), qui réduit fortement la probabilité d'avoir 2 clés identiques

En résumé

Pas de méthodologie bien définie pour modéliser en NoRel :

- ▶ Lien entre les entités par la construction d'une clé "qui inclut une clé" (e.g., review:123:book)
- ▶ Redondance de l'information selon les besoins applicatifs
- ▶ Pas de valeurs nulles, car on n'utilise que les attributs dont on a besoin



Bilan

- ▶ **Big Data + Web** \Rightarrow nouveaux besoins pour la modélisation, le stockage et le traitement de données **volumineuses**, véloces (**flux**) et variées (**hétérogénéité**)
- ▶ **Théorème de CAP** (consistency, availability, partition tolerance)
- ▶ **Mouvement NoSQL / NoRel / NewSQL** :
 - ▶ Entrepôt clé-valeur
 - ▶ BD orientée colonnes
 - ▶ BD orientée graphes
 - ▶ BD orientée documents
 - ▶ ...

Perspective : traitement en parallèle avec le paradigme
Map-Reduce

Des questions ?

