

JDEV - Le retour

Dans la thématique Big Data, quelques mots sur les bases NoSQL, et sur les opérations de type Map-Reduce via l'implémentation Hadoop

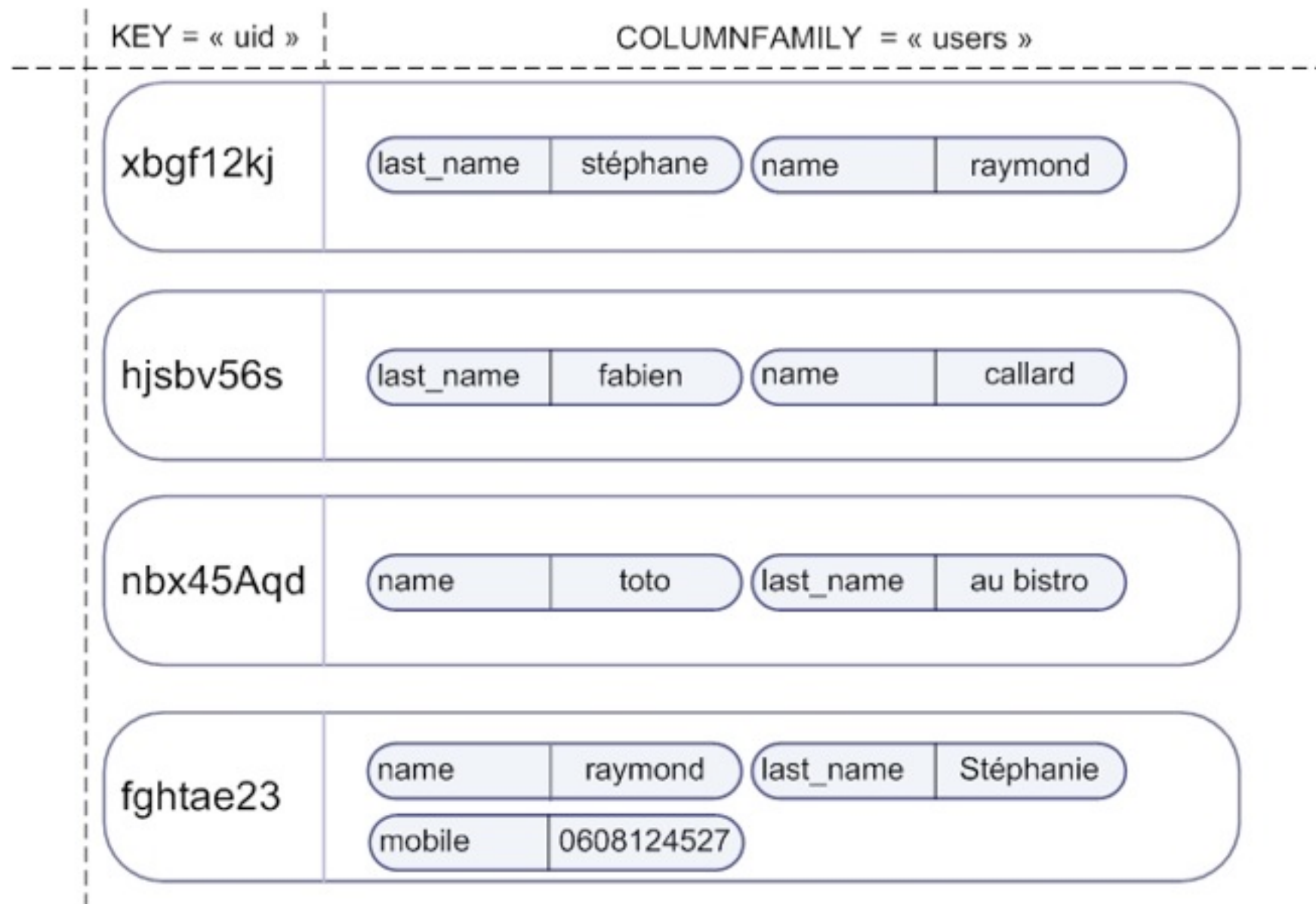
NoSQL

- Not only SQL
- Systèmes de stockage et de manipulation de données non relationnels, non transactionnels
- Dans quel cas ?
 - ▶ Volume de données important, systèmes distribués
 - ▶ Besoin d'un modèle qui peut évoluer facilement
 - ▶ Modèle relationnel non adapté pour décrire les données
 - ▶ ...?

Les types de bases NoSQL

- Clé - valeur (ex : Redis, Oracle NoSQL)
 - **Orientées colonnes (Cassandra, HBase)** - Données du web
 - Orientées document (MongoDB - SolR/Elasticsearch - CouchDB) - émergence du json
- Graphe (Neo4j) - Réseaux sociaux

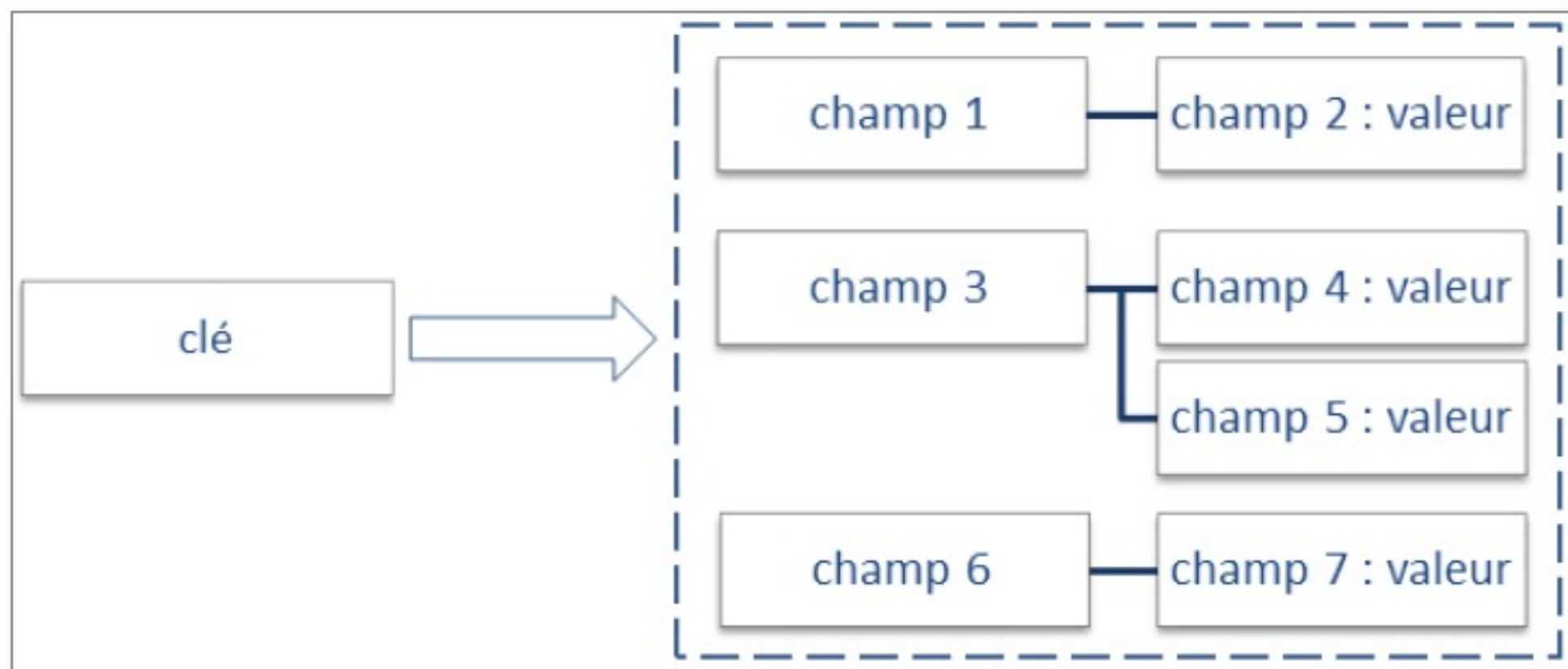
Bases orientées colonne



(site <http://www.stephane-raymond.com/blog/nosql/debuter-avec-cassandra-et-php/>)

Bases orientées colonne

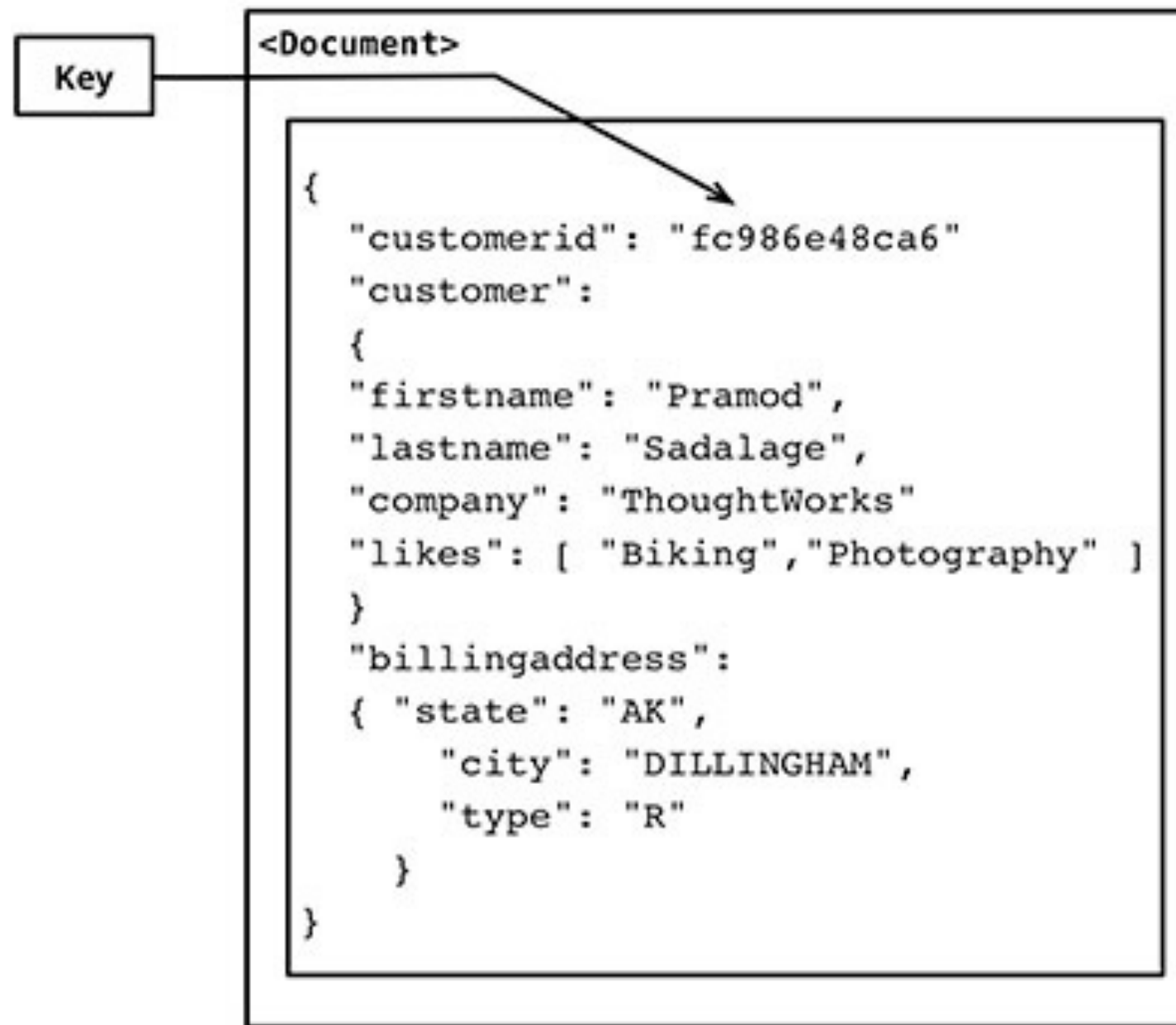
- Evolution du modèle clef-valeur.
- Se rapprochent des base relationnelles, mais organisées en colonnes et non en lignes
- --> Très performantes en lecture (tri, agrégation)
- Couteuses à l'insertion/modification/suppression
- Un exemple là : <http://www-igm.univ-mlv.fr/~dr/XPOSE2010/Cassandra/modele.html>
- Et là : <http://www.stephane-raymond.com/blog/nosql/debuter-avec-cassandra-et-php/>



Bases orientées document

- Aussi une évolution du modèle clé-valeur. La valeur est un document complexe
- Ajout, modification, suppression de seulement certains champs dans un document
- Indexation de champs --> accès rapide sans passer par la clef
- Requêtes plus élaborées
- Emergence du JSON

Bases orientées document



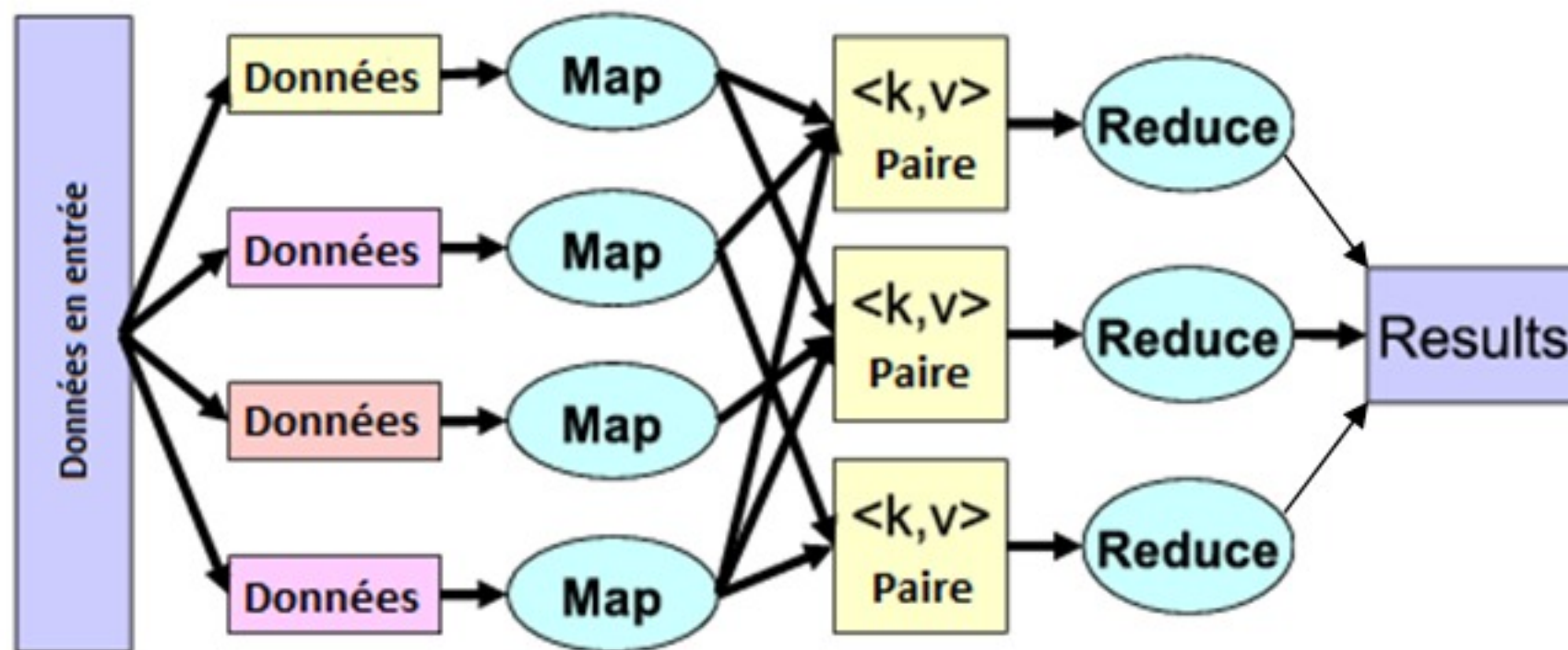
Bases orientées graphe

- Quand les relations entre données sont complexes et représentent le coeur de l'information
- Basées sur la théorie des graphes
- Ex : Réseaux sociaux

Hadoop et Map Reduce

- Hadoop : framework java pour gros volumes. Outils : HDFS, HBase, Hive, MapReduce, ... Fondation apache, inspiré par les travaux google
- Map reduce : Modèle de programmation destiné à la manipulation d'un nombre important de données distribuées. Implémentation la plus connue : Hadoop
- Spark : Autre implémentation de mapReduce, plus performante et un peu plus simple d'utilisation

Hadoop et Map Reduce



Map Shuffle Reduce

Ex (partiel) : word count

```
public class WordCount {

    public static class Map extends MapReduceBase implements Mapper<LongWritable, Text, Text, IntWritable> {
        private final static IntWritable one = new IntWritable(1);
        private Text word = new Text();

        public void map(LongWritable key, Text value, OutputCollector<Text, IntWritable> output, Reporter reporter) throws
IOException {
            String line = value.toString();
            StringTokenizer tokenizer = new StringTokenizer(line);
            while (tokenizer.hasMoreTokens()) {
                word.set(tokenizer.nextToken());
                output.collect(word, one);
            }
        }
    }

    public static class Reduce extends MapReduceBase implements Reducer<Text, IntWritable, Text, IntWritable> {
        public void reduce(Text key, Iterator<IntWritable> values, OutputCollector<Text, IntWritable> output, Reporter reporter)
throws IOException {
            int sum = 0;
            while (values.hasNext()) {
                sum += values.next().get();
            }
            output.collect(key, new IntWritable(sum));
        }
    }

    public static void main ....
    ...
}
```