



enki Multimedia
minimal web

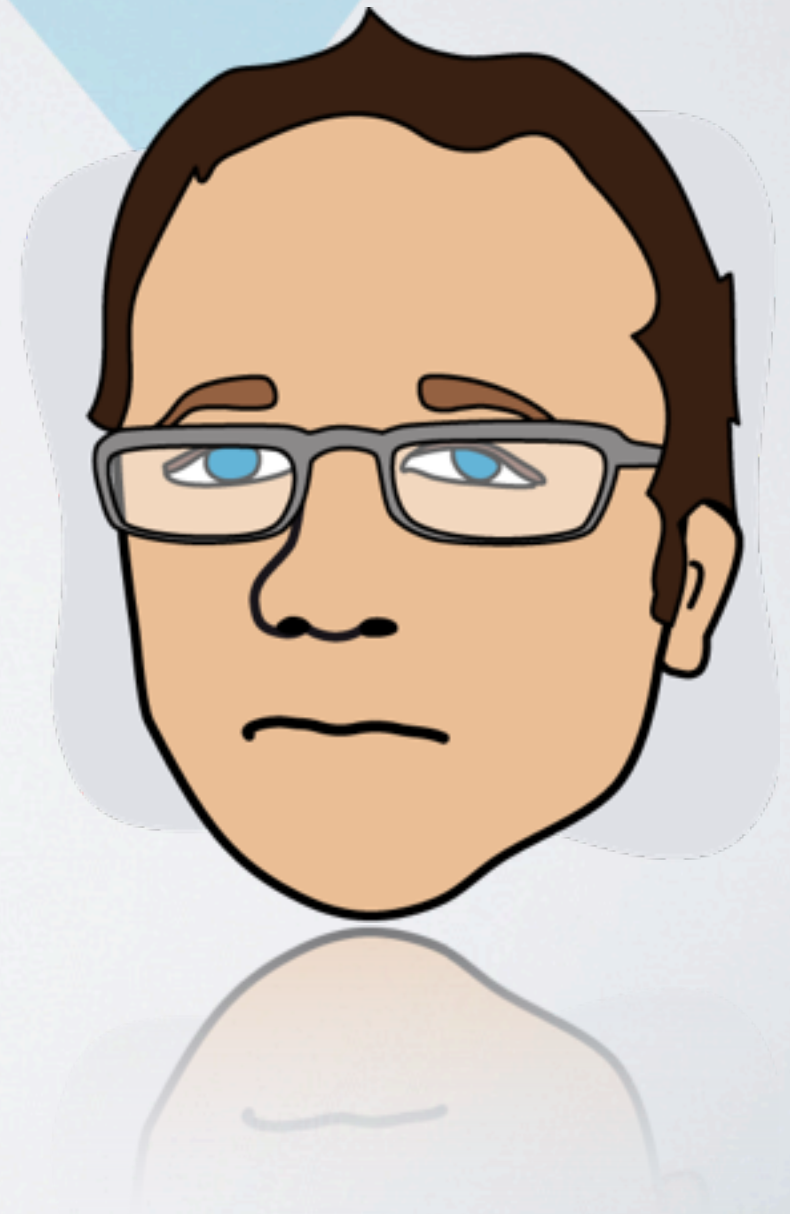
PYTHON ET LES BASES DE DONNÉES NON SQL

Benoît Chesneau
31/05/2009 PYCON FR 2009

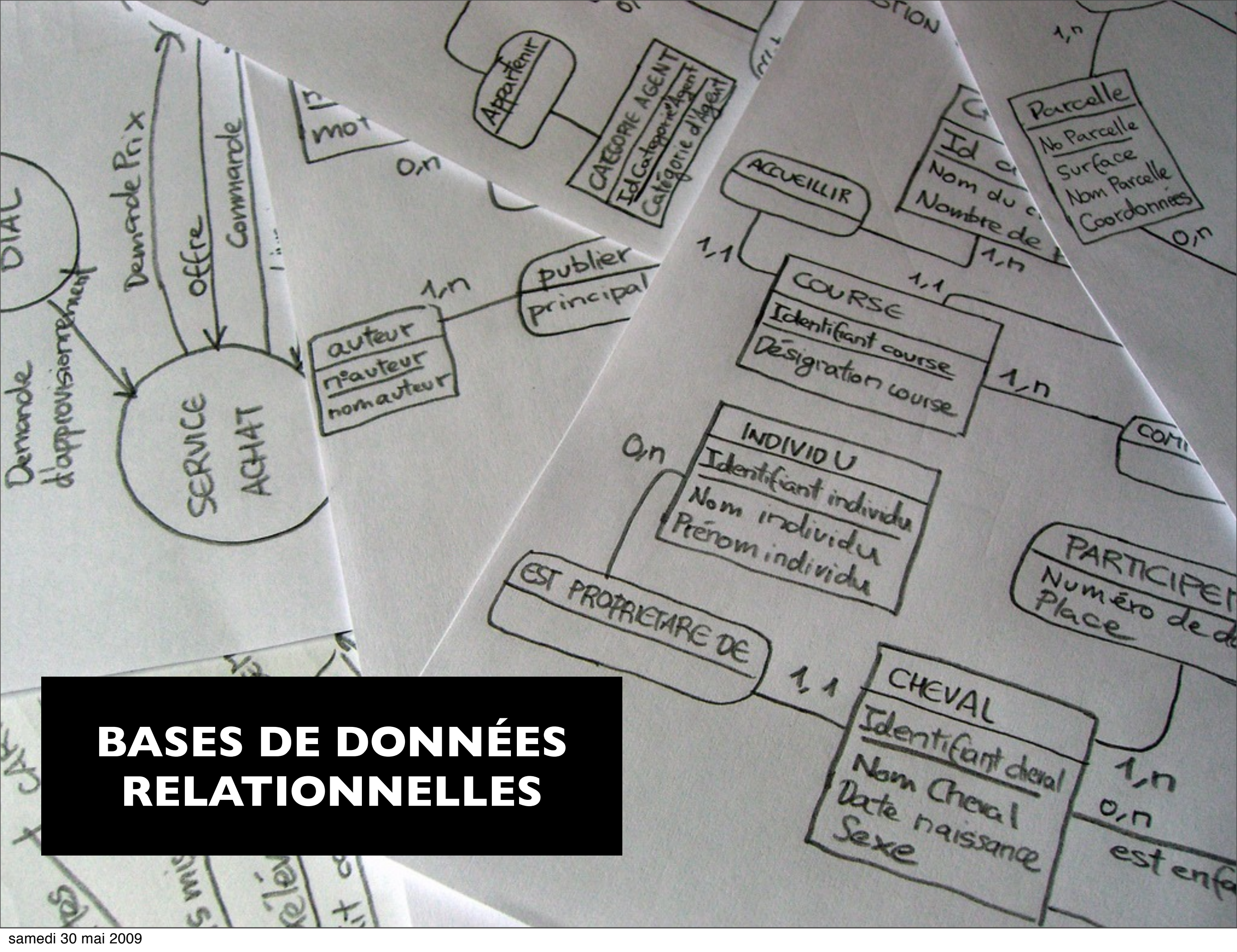
introduction

enki Multimedia minimal web

- ❑ 1998, CanalFood, les menus de restaurants
- ❑ 2008, créé Enki Multimedia
- ❑ minimal web
- ❑ Échanger des ressources



BASES DE DONNÉES RELATIONNELLES





**Faire rentrer des carrés
dans des ronds**

- ❑ Modélisation de données
- ❑ Migration difficile
- ❑ Problèmes de scalabilité



Ça fatigue.

The background is a dynamic, abstract digital composition. It features numerous glowing, semi-transparent cubes in various colors (blue, purple, yellow, green) floating in a dark blue space. These cubes are interconnected by a complex network of thin, white, glowing lines that form a web-like structure. Bright, out-of-focus light points are scattered throughout the scene, creating a sense of depth and energy. The overall aesthetic is futuristic and high-tech.

**DES DONNÉES
VARIÉES ET
TOUJOURS PLUS
NOMBREUSES.**

ACID

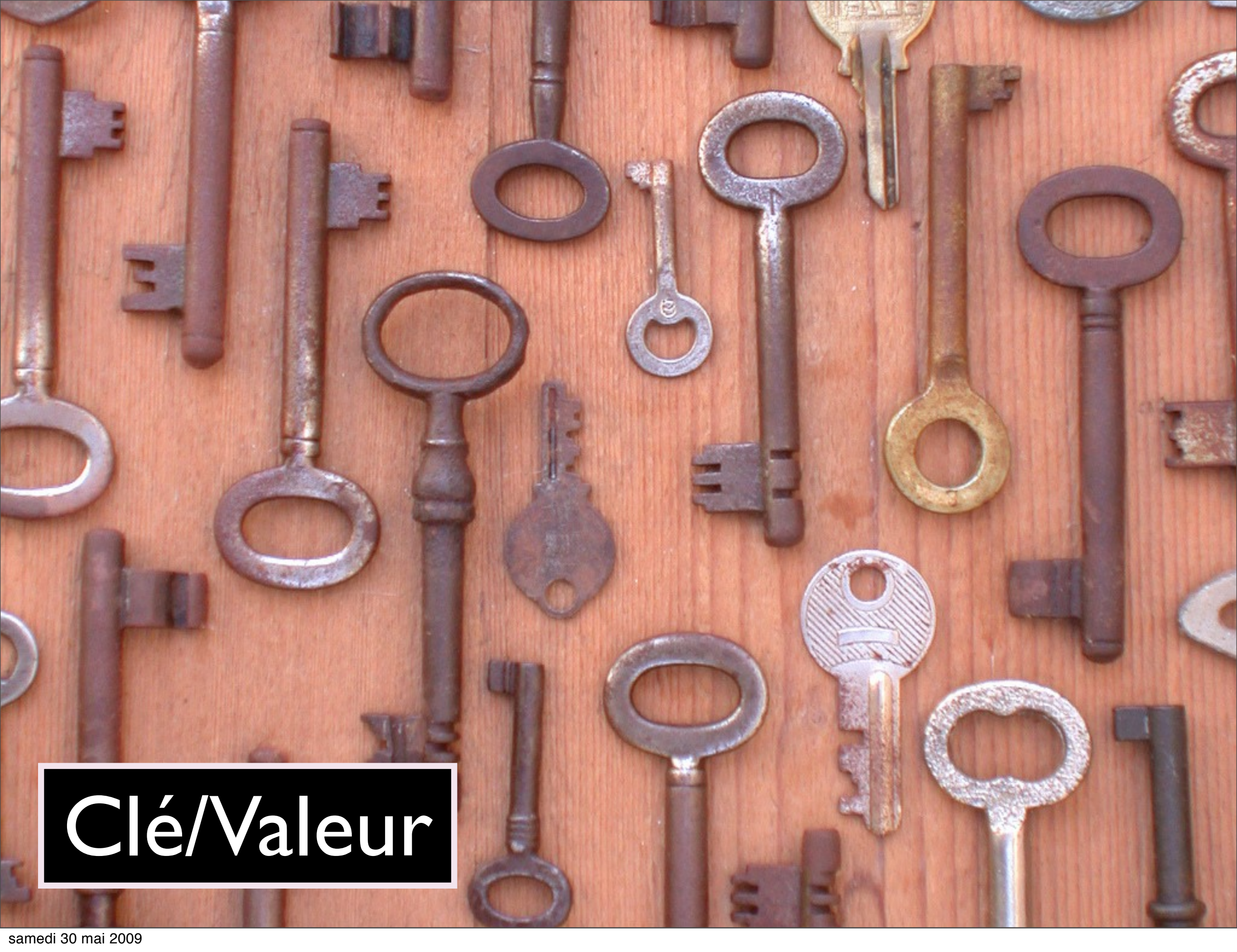
- **Atomicity**: tout ou rien
- **Consistency**: données toujours accessibles
- **Isolation** : distinction écritures/lectures
- **Durability**: pas de mensonges

CAP

- **Consistency:**
- **Availability:**
- **Partition tolerance**

BASE

- **Basically Available**
- **Soft state**
- **Eventually consistent**
- **Tout le monde le fait (google, amazon, facebook...)**



Clé/Valeur

MEMCACHED

- Pas de persistance, Ram seulement
- Rapide
- Utilisé principalement pour du cache
- Tout le monde l'utilise

MEMCACHED

- **python-memcached:** <http://www.tummy.com/Community/software/python-memcached/>
- ...

TOKYO CABINET TYRANT

- Persistent sur le disque
- Performant
- Activement développé
- Système de réplication équivalent à MySQL.
- tokyo distopia & index inversé

TOKYO CABINET TYRANT

- **pytc**: <http://pypi.python.org/pypi/pytc>
- **tc** : <http://github.com/rsms/tc/tree/master>
- **pytyrant**: <http://code.google.com/p/pytyrant/>
- **Lightcloud**: <http://opensource.plurk.com/LightCloud/>

REDIS

- Pas seulement un dépôt clé/value
- Valeurs peuvent être des listes, sets, strings, entiers...
- Possibilité de prendre des ensemble de valeurs
- réplication

REDIS

- Client python officiel
- <http://code.google.com/p/redis/>



BDs orientées Document



COUCHDB

- Réplication asynchrone
- Documents en JSON
- Vues automatiquement incrementées
- map/reduce



COUCHDB

- **couchdb-python:** <http://code.google.com/p/couchdb-python/>
- **couchdbkit:** <http://bitbucket.org/benoitc/couchdbkit/>
- **couchapp:** <http://github.com/couchapp/couchapp/>



- Rapide
- JSON ou BSON (json binaire avec des extensions)
- Réplication asynchrone
- Système d'index
- Système de requête avancé (ad hoc)
- Gestion de références entre documents (nested documents)



- **pymongo** : <http://github.com/mongodb/mongo-python-driver/tree/master>
- **app-engine connector** : <http://github.com/mongodb/mongo-appengine-connector>



BDs orientées colonnes



HBASE

- Sous-projet hadoop. basé sur hfs
- entièrement distribué
- bigtable like
- map/reduce et cascading



HBASE

- Accès en python via jython.
- Thrift

CASSANDRA

- Facebook
- Mélange de Dynamo et BigTable
- éventuellement consistant
- facilité d'administration

CASSANDRA

- Thrift

HYPERTABLE

- <http://hypertable.org/>
- Thrift

BDs objets





- Python object persistence
- problème de la migration
- Gestion des conflits en lecture (écriture?)
- Atomicité
- Scalabilité possible via ZEO et ZRS (payant)

Questions





@benoitc