

FORMATION SQL ESSENTIEL

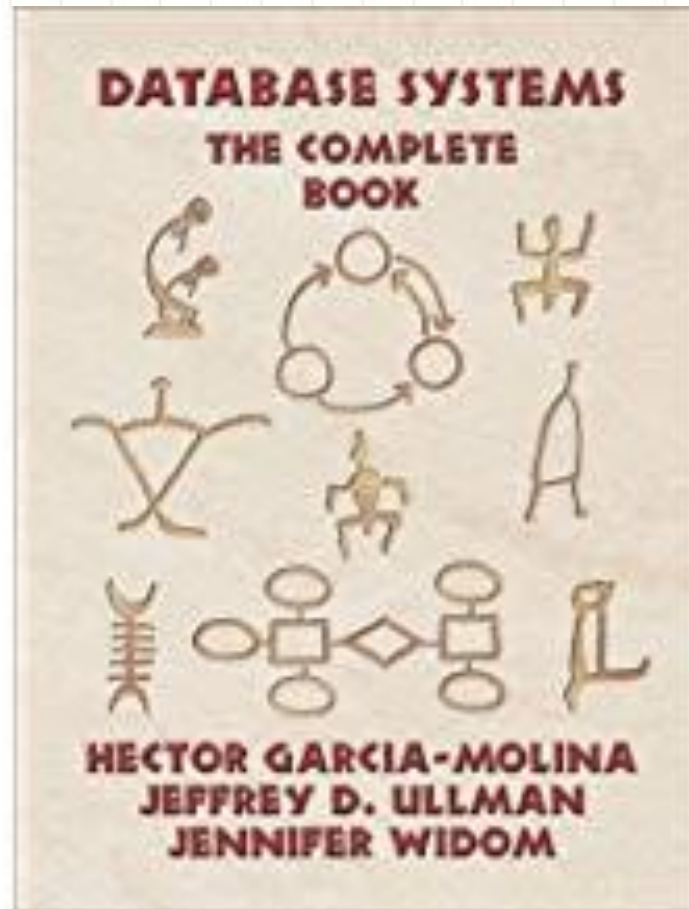
LES RDBMS

Mikaël DAUTREY
(mikael.dautrey@isitix.com)

Plan

- DBMS
- (R)DBMS
- SQL
- ACID

Bibliographie



DBMS

- Stockage persistant des données
- Interface de programmation
 - Ecriture, mise à jour, suppression, lecture
- Gestion de la concurrence

Pourquoi des bases de données ?

“ Bad programmers worry about the code. Good programmers worry about data structures and their relationships.” (Linus Torvalds)

- Accès : Ecrire, mettre à jour, accéder **rapidement et de manière fiable** à la donnée dont le programme a besoin
- Temporalité : Assurer la conservation des données dans le temps, récupérer les données après erreur, revenir en arrière, arrêter un état, ...
- Partage : permettre à plusieurs processus d'accéder aux mêmes données dans le même intervalle de temps de manière cohérente

La question du stockage : du physique au logique

Data locality is less crucial in modern datacenters but is still relevant for WAN distributed architecture

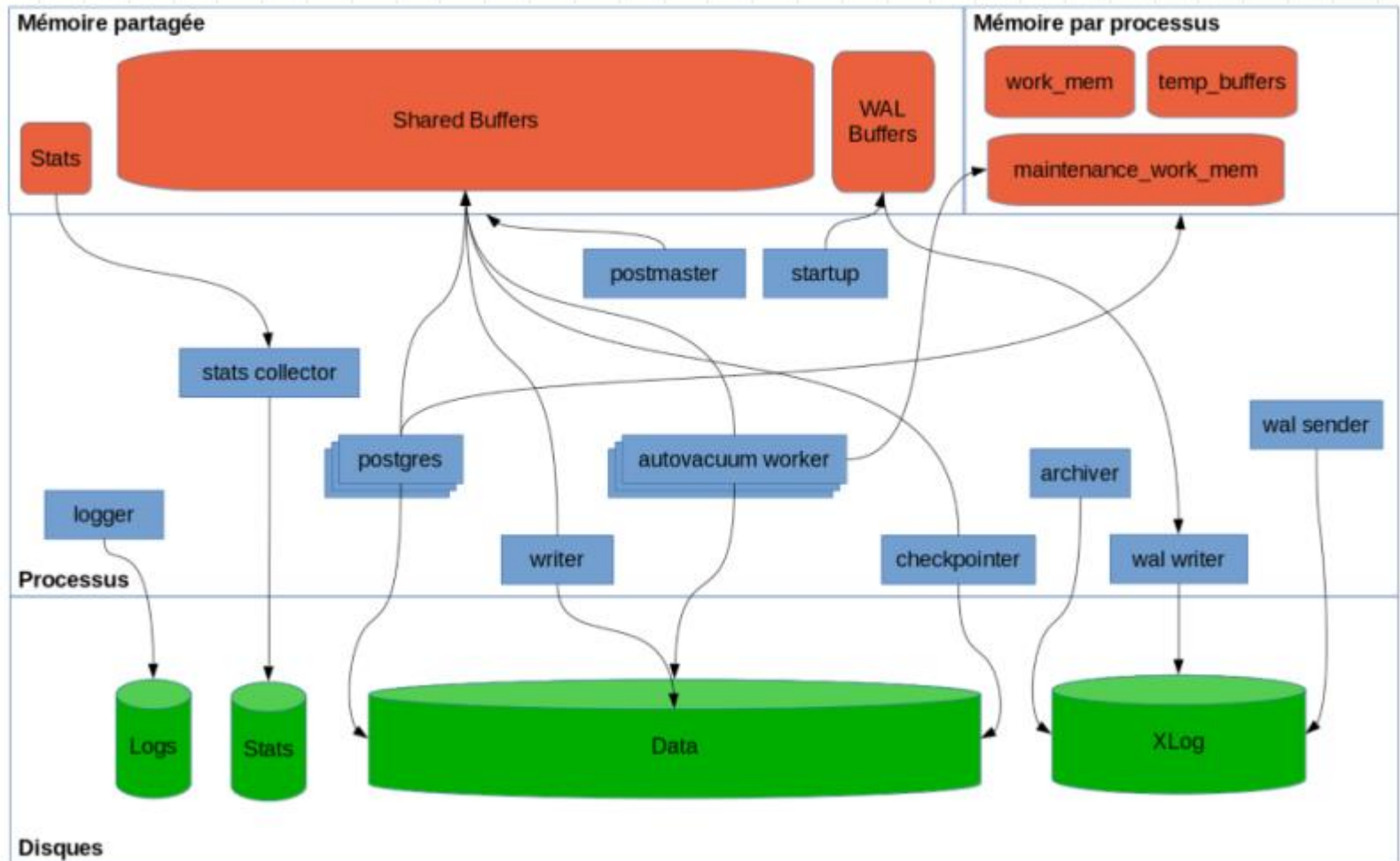
- RAM access time : nanosecond
- SSD disks access time : 0,1 millisecond
- SAS disks access time : several milliseconds
- Low latency switches with high throughput :
 - microseconds and less for latency
 - 10Gb/40Gb of throughput

Low price of RAM, SSDs disks and performance of Ethernet switches → architecture patterns for big data platform and database architecture in general have to be reconsidered

- Capacité
- Performance (IO, temps de réponse)
- Scalabilité
- Fiabilité : réplication, correction d'erreurs...
- Optimisation : compression, déduplication, tiering
- Localisation des données : proximité stockage/traitement

La question de l'accès

Postgresql : architecture et notions avancées (Guillaume Lelarge)



(R)DBMS

“Theory and practice sometimes clash. And when that happens, theory loses. Every single time” (Linus Torvalds)

- Relationnel ?
 - Dépendance fonctionnelle
 - Tuples d'attributs = rangs
 - Schéma fixe
 - Normalisation

(R)DBMS

- Définition d'une algèbre avec des opérations ensemblistes
 - Union, intersection, différence, produit, jointure, theta-jointure, projection, sélection
 - Groupement, renommage
- Ensemble incomplet au sens de Turing
- Mais bon niveau d'expressivité et propriétés algébriques favorables à l'optimisation

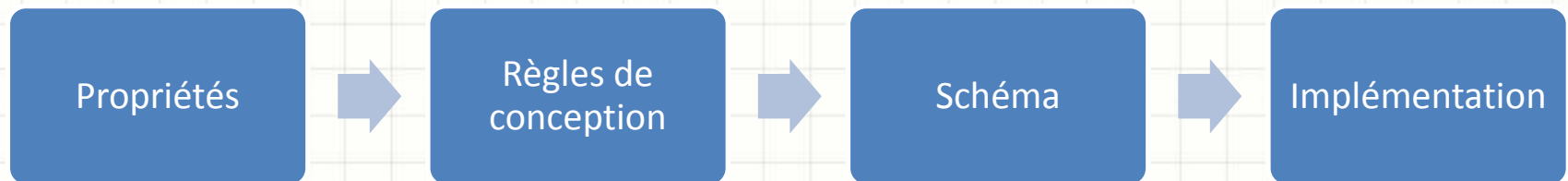
SQL est la traduction adaptée (pratique vs théorie) de cette algèbre ensemble sous forme d'un langage de programmation

SQL

- Un langage ensembliste
 - Modélisation des données
 - Accès (Read, Write, Update, Delete) aux données

Modélisation de données SQL

- Modélisation de données
 - Des propriétés des données à modéliser dépendant du domaine
 - Attributs (domaine des attributs)
 - Clés
 - Relations (relations entre les attributs, entités, relations entre entités)
 - Des règles de conception d'un schéma dépendant de la logique associée à la base de données
 - Relationnel : formes normées (Boyce-Code, 1NF, 2NF, 3NF)

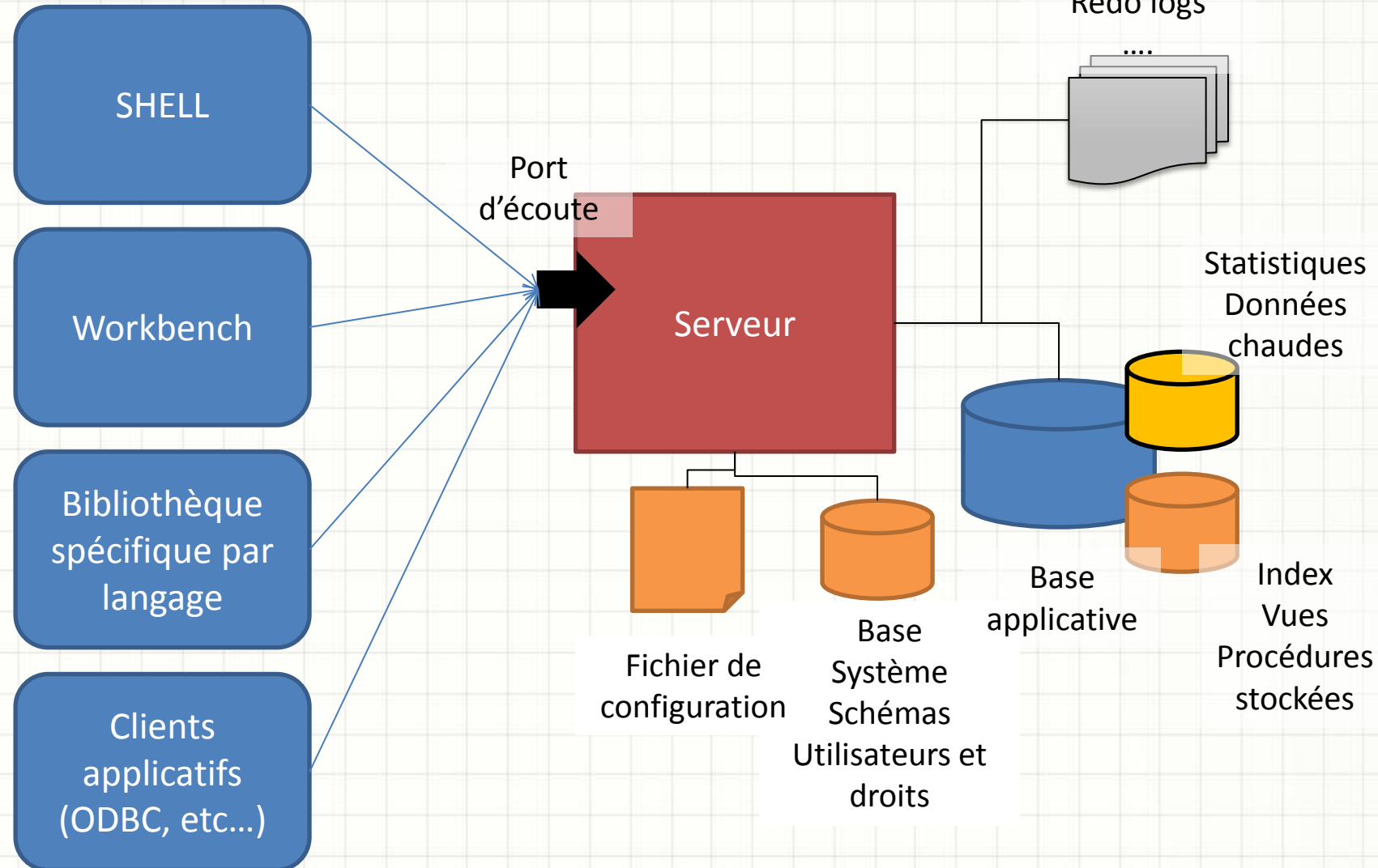


SQL Data Definition Language

Accès aux données SQL

- Un langage et des opérations ensemblistes
 - SELECT
 - INSERT
 - UPDATE
 - DELETE

RDBMS : une architecture client/serveur



Succès de SQL ?

- Fiabilité
- Concision
- Adoption par les utilisateurs
- Polyvalence



**VOUS AVEZ DES
QUESTIONS ?**