Bases de données documentaires et distribuées, http://b3d.bdpedia.fr

Introduction du cours

Le sujet, en bref

Documents, collections massives, stockage et calcul distribués

Représentation d'entités sous forme de **documents** pour la gestion de **collections** à très grande échelle dans des environnements distribués.

Les grands thèmes :

- modélisation sous forme de **document**, structuré, semi-structuré, non structuré ;
- **recherche d'information** dans de très grandes collections ;
- **distribution du stockage** pour maintenir de très grands volumes de données ;
- **distribution des calculs**, notamment pour les méthodes analytiques.

Des mots-clés à comprendre (et à critiquer) : données structurées/non structurées, bases NoSQL, Cloud, BigData, moteurs de recherche, etc.

Point de départ : bases relationnelles

Les **bases relationnelles** = applications gérant des informations **structurées** et **régulières** : applications de "gestion", Web, mobiles.

- Une modélisation normalisée.
- Un langage (SQL) très bien défini, normalisé lui aussi.
- De très bonnes performances, obtenues **automatiquement**.
- Une gestion robuste de la concurrence d'accès.

Attention à bien apprécier ce qu'on gagne / perd en passant au "NoSQL" (on gagne marginalement, on perd beaucoup)

Révisions nécessaires ? Mieux vaut s'en apercevoir **maintenant**. Révisions sur `http://www.bdpedia.fr`

Plan du cours

Quatre parties, 3 semaines chacune, 1 jalon à la fin de chaque partie

- **Représentation de documents textuels** : les formats XML et JSON ; la modélisation de documents, les langages de manipulation.
- **Recherche d'information dans les bases documentaires** : moteurs de recherche, index, algorithmes.
- **Stockage distribué**. Systèmes distribués ; NoSQL.
- **Calcul distribué**. MapReduce, et au-delà

Connaissances pratiques (en cours, en TP, chez vous)

- Des systèmes "NoSQL" ; (MongoDB, CouchDB, Cassandra)
- Des moteurs de recherche (Solr, ElasticSearch).
- Systèmes distribués : MongoDB, ElasticSearch, Cassandra.
- Traitement massif : Hadoop, Spark, Flink

Prérequis

- **Compréhension des bases relationnelles**, soit au moins la conception d'un schéma, SQL, ce qu'est un index et des notions de base sur les transactions.
  $\Rightarrow$ si nécessaire, voir `http://www.bdpedia.fr`
- **Une aisance minimale dans un environnement de développement**. Editer un fichier, lancer une commande, ne pas paniquer devant un nouvel outil, savoir résoudre un problème avec un minimum de tenacité, etc.
  $\Rightarrow$ Vous devez reproduire les exemples donnés.

La connaissance de langage de programmation comme Java ou Python est un plus pour consolider vos nouvelles connaissances.

Environnement matériel et logiciel

Le cours repose beaucoup sur la mise en pratique.

- au Cnam, tout vous est fourni ;
- Cnam à distance, ou en auditeur libre, il vous faut une machine dotée d'au moins 8 GO de mémoire (Linux, Mac/OS ou Windows).

Tous les logiciels utilisés sont libres de droits. Nous travaillons essentiellement avec

- Solr, et ElasticSearch pour l'indexation de documents.
- MongoDB, Cassandra, et plusieurs autres outils NoSQL.
- Flink et Spark.

L'installation de ces outils (et leur utilisation) peut se faire avec **Docker**, un gestionnaire de systèmes distribués **virtuels**.

Comment travailler

Le cours est découpé en **chapitres**, couvrant un sujet bien déterminé, et en **sessions**.

L'unité de travail est la session. Chaque session demande quelques heures de travail personnel

Pour assimiler une session vous pouvez combiner les ressources suivantes :

- **La lecture** du support en ligne, également disponible en PDF ou en ePub.
- **Le suivi du cours**, en vidéo ou en présentiel.
- **Le test des exemples de code** fournis dans chaque session.
- **La réalisation des exercices** proposés en fin de session.

La réalisation des exercices et des manipulations est essentielle.

Mener votre travail personnel

Vous devez maîtriser le contenu des sessions **dans l'ordre où elles sont proposées**.

Commencez par lire le support, jusqu'à ce que les principes vous paraissent clairs.

Reproduisez les exemples de code. Tous les exemples donnés sont testés et doivent donc fonctionner.

Cherchez à résoudre les problèmes par vous-mêmes au besoin : c'est le meilleur moyen de comprendre.
Finissez par les exercices. Faites les vous-mêmes, avant de regarder la solution.
Répondez au Quiz proposé en fin des principaux chapitres.

Evaluation : option 1, le projet

Objectif : un sujet qui vous intéresse, en liaison avec les techniques avancées de gestion de données, et vous préparez un rapport (app. 20 pages). Travail **personnel**, plagiat **éliminatoire**

Exemple : vous étudiez le dernier système NoSQL à la mode (pas celui vu en cours !)

- Une partie **pratique** : choix d'un jeu de données, insertion, manipulations
- Une partie **étude** d'un aspect technique (interrogation, distribution, tolérance aux pannes, ...)

C'est vous qui proposez le sujet ! Site de soumission ouvert
`http://deptfod.cnam.fr/soumissions`

Evaluation : option 2, le hackathon

Objectif : travailler **en groupe** pour mettre en place un **vrai** système distribué à la fin du cours, en une séance.

Thèmes proposés (`http://b3d.bdpedia.fr/hackathon.html`)

- Traitement massif de flux Twitter avec Kafka et Spark
- Une *blockchain* basée sur un système NoSQL
- Autre idée ?

Travail collaboratif

- Réunion du groupe au moins toutes les deux semaines
- Outils comme Slack, Git, environnements virtuels

Il faut constituer les groupes rapidement, travailler en continu.
Rush final à la fin de l'année, présentation en janvier.

Evaluation

Pour les auditeurs du Cnam, l'évaluation de l'UE combine :

- Pour 1/3 :

  Soit une note portant sur le **rapport écrit** du projet (coeff. 1/3).
  Soit une note sur la participation au hackathon (avec présentation orale et rapport)
- La note d'un examen de contrôle (coeff. 2/3).

Projet / hackathon sont suivis et évalués par les enseignants du Cnam.

Voir support en ligne pour les annales, et les instructions détaillées.

Bon travail !