

Technologies du Web

Ludovic DENOYER et Laure SOULIER - laure.soulier@lip6.fr

UPMC

15 mars 2017

Rappel sur le cours

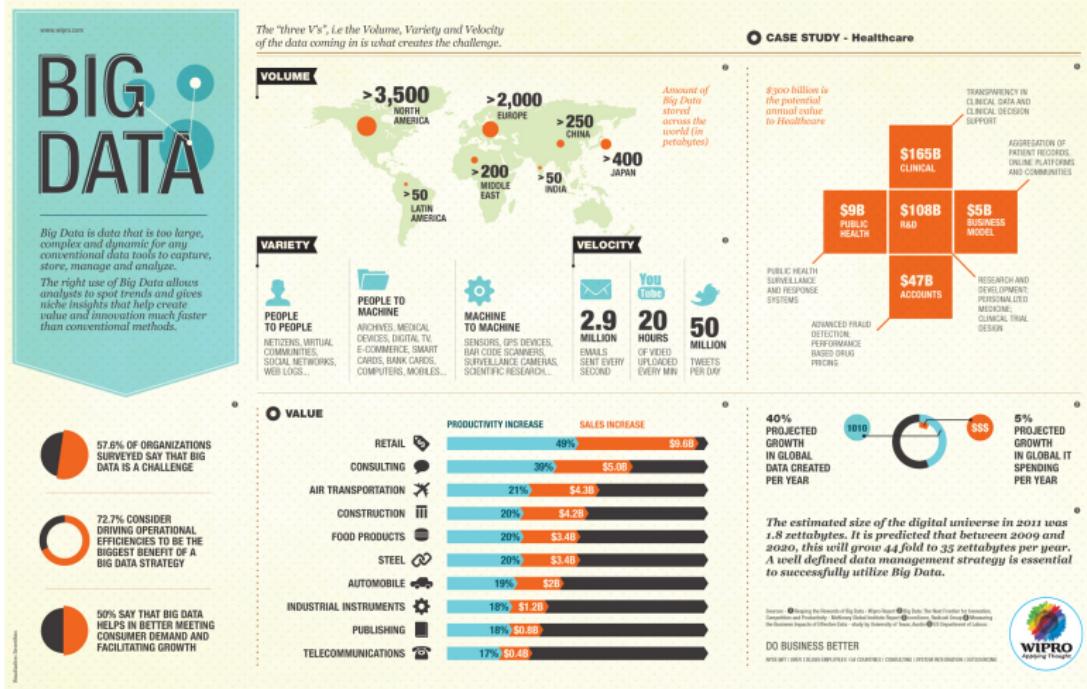
- Soutenances
 - 2 slots de 1h45 heures le mardi 18 avril : 14h et 18h
 - 2 slots de 1h45 heures le mercredi 19 avril : 8h45 et 10h45
 - 2 slots de 1h45 heures le jeudi 20 avril : 8h45 et 18h
- Documents attendus :
 - Le site qui fonctionne SUR LE SERVEUR LI328 (sauf moteur de recherche éventuellement)
 - Un document (une page) décrivant ce que vous avez fait et ce que vous n'avez pas fait (liste de fonctionnalités du site)
 - Une documentation des web services implémentés
 - Des modifications vous seront demandées durant la soutenance.
 - Voir détails qui seront postés sur site Piazza.
- Moteur de Recherche : Implémentation bientôt
- Cours 10 (semaine prochaine) : Examen de l'année dernière (annales) corrigées.

Plan

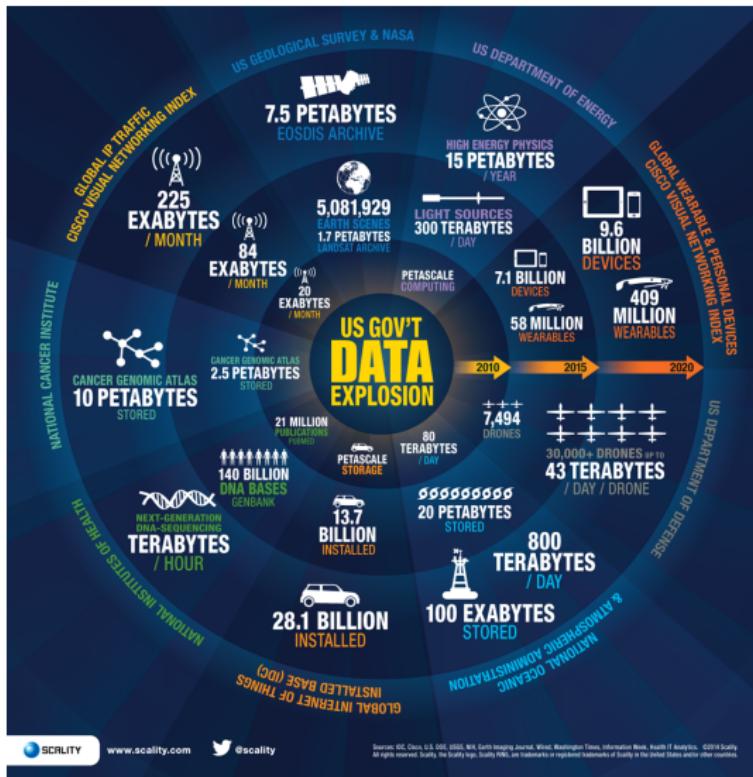
- Contexte
- Map Reduce
- Moteur de Recherche

Contexte

Contexte



Contexte



Data driven science : le 4e paradigme (Jim Gray - Prix Turing)

SNR 2013

Extrait : "A l'heure actuelle, la science vit une révolution qui conduit à nouveau paradigme selon lequel 'la science est dans les données', autrement dit la connaissance émerge du traitement des données [...] **Le traitement de données et la gestion de connaissances représentent ainsi le quatrième pilier de la science après la théorie, l'expérimentation et la simulation.** L'extraction de connaissances à partir de grands volumes de données (en particulier quand le nombre de données est bien plus grand que la taille de l'échantillon) , l'apprentissage statistique, l'agrégation de données hétérogènes, la visualisation et la navigation dans de grands espaces de données et de connaissances sont autant d'instruments qui permettent d'observer des phénomènes, de valider des hypothèses, d'élaborer de nouveaux modèles ou de prendre des décisions en situation critique"

Traitement de données

.....
68% des entreprises qui ont systématiquement recours à une analyse de données dans leurs prises de décision voient leurs bénéfices augmenter

* *selon une étude menée par the Economist Intelligence Unit (2014)*

.....
Pour qui réussit à optimiser son usage, la donnée devient **information**, puis, bien partagée au sein de l'entreprise, elle se transforme en **connaissance** et constitue son **savoir**. Elle peut être une source de services et d'innovations, notamment lorsqu'on la croise avec d'autres données et qu'elle provient de sources diverses.

* *Enjeux Business des données - CIGREF 2014*

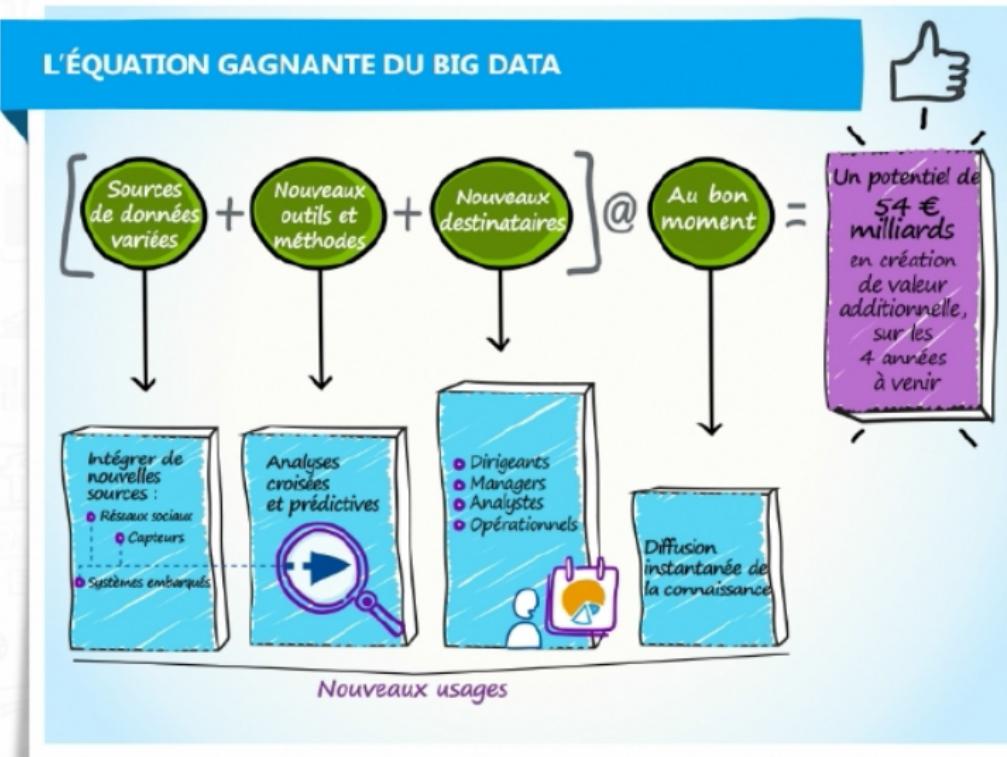
Traitement de données en entreprise

...

La donnée est donc l'un des principaux actifs immatériels de nos organisations, et pour autant, n'est pas encore gérée avec la même rigueur ni les mêmes moyens que les autres ressources, capital et ressources humaines notamment. Dans un contexte où elle est devenue critique pour l'activité de l'entreprise, la mise en place d'une gestion structurée et industrielle de la donnée est impérative.

* *Enjeux Business des données - CIGREF 2014*

Etude IDC - Microsoft 2014



Etude IDC - Microsoft 2014

ÉVOLUTION DU CUMUL DE VALEURS DE LA DONNÉE (en milliards d'euros)



● Entreprises aux usages standards

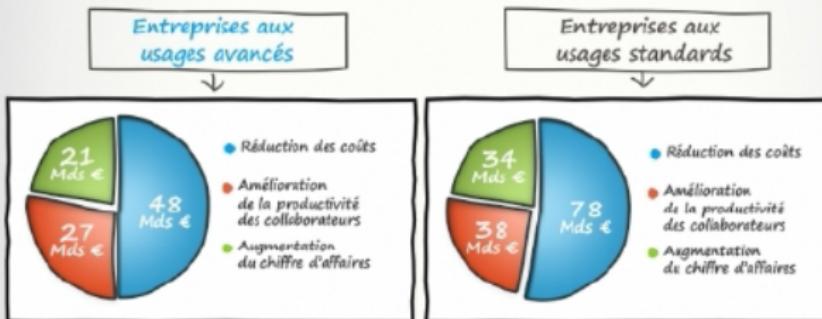
● Entreprises aux usages avancés

Etude IDC - Microsoft 2014

COMPÉTITIVITÉ DES ENTREPRISES ET DONNÉES



Une meilleure exploitation des données pourrait améliorer la compétitivité des entreprises à plusieurs niveaux



Dans ce cours

Calcul distribué

Un des enjeux concerne le traitement de grandes quantités de données. Ce traitement ne peut être réalisé avec les paradigmes classiques de traitement de données et nécessite l'utilisation de plateformes distribuées de calcul

Map Reduce

Distribution des Données

Contexte

- Très grand Volume de Données
 - Google = 20 milliards de pages Web = 400 TeraOctets, 30-35 MB/sec (lecture sur disque) ⇒ 4 mois de lecture.
- Données Distribuées

Problématique

Comment effectuer des calculs sur ces grandes masses de données ?

Problématique

Comment effectuer des calculs sur ces grandes masses de données ?

Pourquoi faire ?

- Indexation
 - Moteurs de Recherche
 - Indexation d'images
- Reporting :
 - Caractériser les utilisateurs d'un réseau social
 - Déetecter la fraude à la CB
 - Statistiques sur des logs de connexions (publicité)
- Analyse :
 - Détection de Communautés
 - Analyse du Churn

Map Reduce

Problématique

Comment effectuer des calculs sur ces grandes masses de données ?

Map-Reduce

- Map-Reduce a été introduit par Google en 2004
- Map-Reduce est un :
 - Un modèle de programmation,
 - avec un schéma très contraint,
 - qui permet :
 - parallélisation automatique,
 - de l'équilibrage de charge,
 - des optimisations sur les transferts disques et réseaux,
 - de la tolérance aux pannes

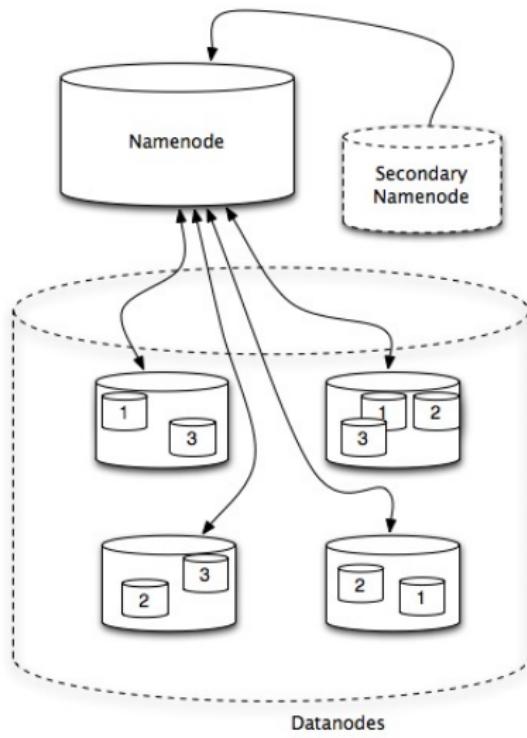
Map-Reduce

- The Yahoo ! Search Webmap is a Hadoop application that runs on a more than 10,000 core Linux cluster and produces data that is now used in every Yahoo ! Web search query.
- Google : the size of one phase of the computation [of the index] dropped from approximately 3800 line of C++ code to approximately 700 lines when expressed using MapReduce.
- Facebook has multiple Hadoop clusters deployed now - with the biggest having about 2500 cpu cores and 1 PetaByte of disk space. We are loading over 250 gigabytes of compressed data (over 2 terabytes uncompressed) into the Hadoop file system every day and have hundreds of jobs running each day against these data sets.
- Hadoop est aussi utilisée par Twitter, Amazon, Rackspace, LinkedIn, IBM, Veoh, Last.fm (en Bash !!), Microsoft...

Traitement distribué

- Apache Hadoop
 - Framework distribué
 - Utilisé par de très nombreuses entreprises
 - Traitements parallèles sur des clusters de machines
 - ⇒ Amener le code aux données
- Système de fichiers HDFS
 - Système de fichiers virtuel de Hadoop
 - Conçu pour stocker de très gros volumes de données sur un grand nombre de machines
 - Permet l'abstraction de l'architecture physique de stockage
 - RéPLICATION des données

Hadoop Distributed File System



Hadoop Distributed File System

- Architecture de machines HDFS

- NameNode :

Gère l'espace de noms, l'arborescence du système de fichiers et les métadonnées des fichiers et des répertoires

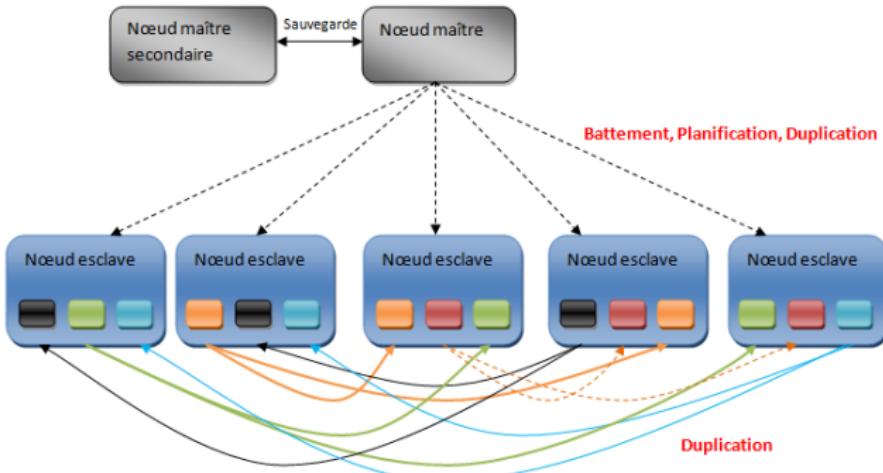
- SecondaryNameNode :

Gère l'historique des modifications dans le système de fichiers
Permet la continuité du fonctionnement du cluster en cas de panne du NameNode principal

- DataNode :

Stocke et restitue les blocs de données.

Hadoop Distributed File System



Hadoop

- De nombreux outils basés sur Hadoop
 - MapReduce : Outil de mise en oeuvre du paradigme de programmation parallèle du même nom
 - HBase : Base de données distribuée disposant d'un stockage structuré pour les grandes tables
 - Hive : Logiciel d'analyse de données (initialement développé par Facebook) permettant d'utiliser Hadoop avec une syntaxe proche du SQL
 - Pig : Logiciel d'analyse de données (initialement développé par Yahoo !) comparable à Hive mais utilisant le langage Pig Latin
 - Spark : Framework de traitement de données distribué avec mémoire partagée
- Plateforme d'apprentissage Mahout

Qui utilise Hadoop ?

YAHOO!

ebay

facebook



Massachusetts
Institute of
Technology

Google

amazon.com



LinkedIn

 **Microsoft**

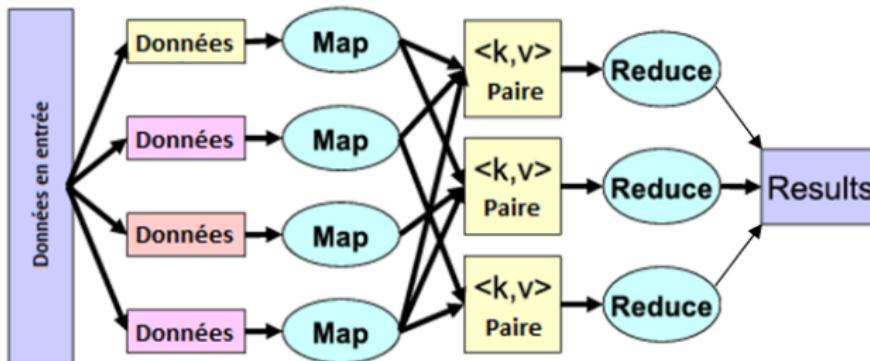
Berkeley
UNIVERSITY OF CALIFORNIA

... et des centaines d'entreprises et universités à travers le monde.

Map Reduce

- Exécution d'un problème de manière distribuée
 - ⇒ Découpage en sous-problèmes
 - ⇒ Execution des sous-problèmes sur les différentes machines du cluster
 - Stratégie algorithmique dite du *Divide and Conquer*
- Map Reduce
 - Paradigme de programmation parallèle visant à généraliser les approches existantes pour produire une approche unique applicable à tous les problèmes.
 - Origine du nom : langages fonctionnels
 - Calcul distribué : "MapReduce : Simplified Data Processing on Large Clusters" [Google,2004]

Map Reduce



- Deux étapes principales :
 - Map : Emission de paires $\{clé, valeur\}$ pour chaque donnée d'entrée lue
 - Reduce : Regroupement des valeurs de clé identique et application d'un traitement sur ces valeurs de clé commune

Map Reduce

- Ecrire un programme Map Reduce :
 - ① Choisir une manière de découper les données afin que Map soit parallélisable
 - ② Choisir la clé à utiliser pour notre problème
 - ③ Écrire le programme pour l'opération Map
 - ④ Écrire le programme pour l'opération Reduce

Map Reduce : WordCount

- Exemple classique : le Comptage de mots
 - Fichiers d'entrée textuels
 - On veut connaître le nombre d'occurrences de chacun des mots dans ces fichiers
- Il faut décider :
 - De la manière dont on découpe les textes
 - Des couples <clé,valeur> à émettre lors du Map appliqué à chaque morceau de texte
 - Du traitement à opérer lors du regroupement des clés communes (Reduce)

Map Reduce : WordCount

Fichier d'entrée :

Celui qui croyait au ciel
Celui qui n'y croyait pas
[...]
Fou qui fait le délicat
Fou qui songe à ses querelles

(Louis Aragon, *La rose et le Réséda*, 1943, fragment)

- Pour simplifier, on retire tout symbole de ponctuation et caractères spéciaux. On passe l'intégralité du texte en minuscules.

Map Reduce : WordCount

Découpage des données d'entrée : par exemple par ligne

celui qui croyait au ciel

celui qui ny croyait pas

fou qui fait le delicat

fou qui songe a ses querelles

- Ici, 4 unités de traitement après découpage

Map Reduce : WordCount

- Opération map :

- Séparation de l'unité en mots (selon les espaces)
- Émission d'une paire <mot,1> pour chaque mot

celui qui croyait au ciel	→	(celui;1) (qui;1) (croyait;1) (au;1) (ciel;1)
celui qui ny croyait pas	→	(celui;1) (qui;1) (ny;1) (croyait;1) (pas;1)
fou qui fait le delicat	→	(fou;1) (qui;1) (fait;1) (le;1) (delicat;1)
fou qui songe a ses querelles	→	(fou;1) (qui;1) (songe;1) (a;1) (ses;1) (querelles;1)

Map Reduce : WordCount

- Après le map : regroupement (ou shuffle) des clés communes
 - Effectué par un tri distribué
 - Pris en charge de manière automatique par Hadoop

(celui;1) (celui;1)

(qui;1) (qui;1) (qui;1) (qui;1)

(croyait;1) (croyait;1)

(au;1) (ny;1)

(ciel;1) (pas;1)

(fou;1) (fou;1)

(fait;1) (le;1)

(delicat;1) (songe;1)

(a;1) (ses;1)

(querelles;1)

Map Reduce : WordCount

- Opération Reduce :

- Sommation des valeurs de toutes les paires de clé commune
- Ecriture dans un (ou des) fichier(s) résultats

```
qui: 4
celui: 2
croyait: 2
fou: 2
au: 1
ciel: 1
ny: 1
pas: 1
fait: 1
[...]
```

Etude de Cas

```
Map(String input_key, String input_values) :  
    foreach word w in input_values:  
        EmitIntermediate( w, "'1'" );
```

```
Reduce (String key, Iterator intermediate_values):  
    int result=0;  
    foreach v in intermediate_values:  
        result += ParseInt( v );  
    Emit( key, String( result ) );
```

Implémentations

- Plusieurs implémentations existent.
- Dans différents langages (C++, C#, Erlang, Java, Python, Ruby, R, ...)
- La plus connue est Hadoop (de la fondation Apache)

Hadoop

```
1 public static class TokenizerMapper extends Mapper<Object, Text, Text, IntWritable>
2 {
3     private final static IntWritable one = new IntWritable(1);
4     private Text word = new Text();
5
6     public void map(Object key, Text value, Context context)
7             throws IOException, InterruptedException
8     {
9         StringTokenizer itr = new StringTokenizer(value.toString());
10        while (itr.hasMoreTokens())
11        {
12            word.set(itr.nextToken());
13            context.write(word, one);
14        }
15    }
16 }
```

Hadoop

```
1 public static class IntSumReducer extends Reducer<Text, IntWritable, Text, IntWritable>
2 {
3     private IntWritable result = new IntWritable();
4
5     public void reduce(Text key, Iterable<IntWritable> values, Context context)
6             throws IOException, InterruptedException
7     {
8         int sum = 0;
9         for (IntWritable val : values)
10        {
11            sum += val.get();
12        }
13        result.set(sum);
14        context.write(key, result);
15    }
16 }
```

Map/Reduce MongoDB

L'implémentation de Map/Reduce sur MongoDB se fait par l'intermédiaire de fonctions programmées en **javascript**

Syntaxe

```
db.runCommand(  
  { mapreduce : <collection>,  
    map : <mapfunction>,  
    reduce : <reducefunction>  
    [, query : <query filter object>]  
    [, sort : <sorts the input objects using this key.  
      Useful for optimization, like sorting by the emit key for fewer reduces>]  
    [, limit : <number of objects to return from  
      collection, not supported with sharding>]  
    [, out : <see output options below>]  
    [, keepTemp: <true|false>]  
    [, finalize : <finalizefunction>]  
    [, scope : <object where fields go into javascript global scope >]  
    [, jsMode : true]  
    [, verbose : true]  
  } )
```

MongoDB implémente une version incrémentale permettant de traiter les collections qui évoluent rapidement (pas étudié ici).

Exemple (documentation MongoDB)

Collection

```
{ time : <time>, user_id : <userid>, type : <type> }
```

Problème

On veut extraire les utilisateurs qui ont au moins un événement de type "sale", et compter leur nombre d'occurrence (par utilisateur).

Exemple (documentation MongoDB)

Solution

```
m = function() { emit(this.user_id, 1); }
>r = function(k,vals) {
...     var sum=0;
...     for(var i in vals) sum += vals[i];
...     return sum;
... }
> res = db.events.mapReduce(m, r,
    { query : {type:'sale'} });
> // or in v1.8+:
> // res = db.events.mapReduce(m, r,
    { query : {type:'sale'} },
    out : 'example1');
```

WordCount sur MongoDB

```
function wordMap() {  
  
    // try find words in document text  
    var words = this.text.match(/\w+/g);  
  
    if (words === null) {  
        return;  
    }  
  
    // loop every word in the document  
    for (var i = 0; i < words.length; i++) {  
        // emit every word, with count of one  
        emit(words[i], { count : 1 });  
    }  
}
```

WordCount sur MongoDB

```
function wordReduce(key, values) {
    var total = 0;
    for (var i = 0; i < values.length; i++) {
        total += values[i].count;
    }
    return { count : total };
}
```

Map Reduce en MongoDB depuis JAVA

Considérons le programme MR suivant :

```
function m() {
    key = typeof( this._id ) == "number" ? this._id : this._id.getYear() + 1900;
    emit( key, { count: 1, sum: this.bc10Year } );
}

function r( year, values ) {
    var n = { count: 0, sum: 0 }
    for ( var i = 0; i < values.length; i++ ){
        n.sum += values[i].sum;
        n.count += values[i].count;
    }

    return n;
}

function f( year, value ){
    value.avg = value.sum / value.count;
    return value;
}
```

Comment exécuter ce programme en JAVA ?

Map Reduce en MongoDB depuis JAVA

```
1 String m = "function() { key = typeof( this._id ) == \"number\" ? " +
2     " this._id : this._id.getYear() + 1900;" +
3     "emit( key, { count: 1, sum: this.bci0Year } );";
4
5 String r = "function( year, values ) { var n = { count: 0, sum: 0};" +
6     " for ( var i = 0; i < values.length; i ++ )" +
7     "{ n.sum += values[i].sum; " +
8     " n.count += values[i].count; } return n; }";
9
10 String f = "function( year, value ) { value.avg = value.sum / value.count; return value; }
```

Puis :

```
1 MapReduceOutput out = coll.mapReduce(m, r, null, MapReduceCommand.OutputType.INLINE, null);
2
3 for ( DBObject obj : out.results() ) {
4     System.out.println( obj );
5 }
```

- **MapReduceCommand.OutputType** peut être **REPLACE** ou **INLINE**

Map Reduce en MongoDB depuis JAVA

```
1 MapReduceCommand cmd = new MapReduceCommand( coll, m, r, null,
2                                         MapReduceCommand.OutputType.INLINE, null);
3
4 cmd.setFinalize(f);
5
6 out = coll.mapReduce(cmd);
7
8 for ( DBObject obj : out.results() ) {
9     System.out.println( obj );
10 }
```

Moteurs de recherche

Moteurs de Recherche

Définition (Wikipédia)

Un moteur de recherche est une application permettant de retrouver des ressources (pages web, articles de forums Usenet, images, vidéo, fichiers, etc.) associées à des mots quelconques

- L'ensemble des mots est appelé **requête**.

Exemple : Moteur de Recherche du Web

Trois étapes :

- Le Crawl : chercher les documents
- L'indexation : formatter les documents
- La recherche : répondre à des requêtes (rapidement, et efficacement)

Moteurs de Recherche

ordinateur portable X Rechercher

ordinateur portable
ordinateur
ordre des architectes
ordre des medecins
ord En savoir plus

Environ 8 320 000 résultats (0,12 secondes) Recherche avancée

[Ordinateur portable](#) Liens commerciaux
www.wellpack.fr/PC-portables Découvrez nos meilleures offres achetez votre pc portable à crédit!

[Ordinateur portable](#)
www.toshiba-europe.com/fr Une gamme PC portables pour toutes les utilisations: Trouvez le vôtre!

[Nouveaux PC portables HP](#)
www.hp.com/fr/intel Bénéficiez d'une remise de 100€ sur un PC équipé d'un Processeur Intel

[Ordinateur Portable - achat/vente Ordinateur Portable - RueDuCommerce](#) ★
Rueducommerce : catégorie **Ordinateur Portable** ordinateurs.
www.rueducommerce.fr/Ordinateurs/3-Ordinateur-Portable/ - En cache - Pages similaires

[PC portables à prix discount – Ordinateur, PC portable pas cher ...](#) ★
PC portables à prix discount : Portables ACER discount, ordinateur portable HP moins cher, toutes les grandes marques et accessoires, extension de garantie ...
www.cdiscount.com/.../ordinateurs...portables/v-10709-10709.html -
En cache - Pages similaires

Moteurs de Recherche

[exlead](#) Personnaliser cette page d'accueil | [Identifiez-vous](#) | [Préférences](#)

Web | Images BETA | [Wikipedia](#) BETA | [Plus](#)

puma Tout Wikipédia Pages en français

Recherche Wikipedia Recherche avancée

Résultats 1-10 sur environ 1 898 pour puma

Vue : [Liste](#) [Tableau](#)

Puma

Le puma (Puma concolor) appartient à la famille des félidés. Puma concolor est la seule à espèce du genre Puma. C'est un gros chat sauvage que l'on peut ...
fr.wikipedia.org/wiki/Puma • [Ajouter aux raccourcis](#)

Catégories : Espèce menacée, Mammifère (nom vernaculaire), plus...
Personnes : Farrell Jackson, Géraldine Véron, plus...
Organisations : PUMA, Cougar Fund
Lieux : Amérique du Nord, Vérone, plus...


[Afficher](#)

Puma (hélicoptère)

quelques exemplaires de l'ATAL Le SA 330 Puma est un hélicoptère militaire de transport moyen franco-britannique biturbine.
[fr.wikipedia.org/wiki/Puma_\(hélicoptère\)](http://fr.wikipedia.org/wiki/Puma_(hélicoptère)) • [Ajouter aux raccourcis](#)

Catégories : Hélicoptère, Avion militaire de la guerre froide, plus...
Organisations : PUMA, Armée de l'air
Lieux : Avon, Roumanie, plus...


[Afficher](#)

Super Puma

hélicoptère est une version améliorée du Puma (son équivalent civil est le Super Puma) Sa cellule peut-être en version courte ou allongée. Il peut être ...
fr.wikipedia.org/wiki/Super_Puma • [Ajouter aux raccourcis](#)

Catégorie : Hélicoptère
Organisations : PUMA, Eurocopter Ag


[Afficher](#)

Puma (équipementier)

PUMA AG Rudolf Dassler Sport (PUMA) est une grande multinationale allemande produisant des chaussures de sport et autres vêtements sportifs ainsi que des ...
[fr.wikipedia.org/wiki/Puma_\(équipementier\)](http://fr.wikipedia.org/wiki/Puma_(équipementier)) • [Ajouter aux raccourcis](#)

Catégorie : Équipementier sportif
Personnes : Rudolf Dassler, Pétr...
Organisations : PUMA, Adidas, plus...
Lieux : Munich, Mexique, Amérique


[Afficher](#)

Préciser la recherche



Adidas Aérospatiale Aérospatiale

Gazelle Aérospatiale Puma Air Force Air forces Aire protégée de l'IUCN - catégorie II Alex Shelyey

Argentina BELL Boeing Canada Chris Sabin Christopher Daniels Canoë-puma Coupe du monde de football 2006 Deutscher Deutschland EC 135 Equipeur sportif Eurocopter Super Puma Europe Falcon Fania Felidae Felidi Fellines FORD Ford Puma FOX France Genius Puma Germany Hélicoptère de transport Infantry fighting vehicles José Luis Rodríguez Joueur argentin de rugby à XV Katzen Living people Manco Cápac Mann Marvel Comics NATO Parques Nacionales de Argentina Peter Puma Peter Parker Petey Williams Puma concord RAF Royal Navy Schauspieler Super Puma Turismo en Argentina UK United States US USA Warner Windows games

• Catégories • Termes associés • Personnes
• Lieux • Organisations

Moteurs de Recherche

Google™

YAHOO! msn®

AOL search Lycos

About.com alltheweb

dogpile excite altavista



LookSmart

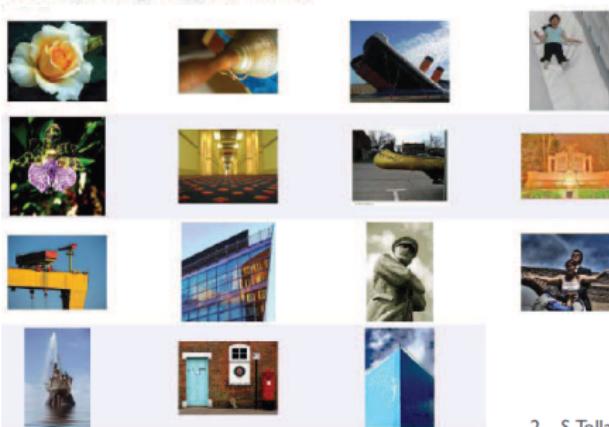


Moteurs de Recherche

<http://www.behold.cc>

beholdTM
find images tagged with
that look like a picture of (a)
and are free to use [] [and modify [] [commercially []]
Note: Image owners can change the licenses of their images after Behold indexes them.
You can safeguard your use of free images from future license changes with [Image Stamper](#).
[Please use images legally. Read our disclaimer & terms of use here.](#)

Searching for images tagged with **Titanic**



Moteurs de Recherche



find images tagged with
that look like a picture of (a)

Titanic
 [?]

and are free to use [and modify] [commercially]

<http://www.behold.cc>

[search](#)

Note: Image owners can change the licenses of their images after Behold indexes them.

You can safeguard your use of free images from future license changes with [ImageStamp](#).

Please use images legally. Read our [disclaimer & terms of use here](#).

Searching for images tagged with **Titanic** that look like **boat**

Behold is using [computer vision](#) to search for images looking like boat



Moteurs de Recherche

GazoPa_{beta}
similar image search

Created: 1 minute ago
Size : 400x300

Search

Options : Shape Any size Any time Gray scale only Omit same Reset parameters Safe search is on

All Video News Sports Twitter Funny Hide

Results 1 - 30 of 1000 for key image

Navigation icons: back, forward, search, etc.

Moteurs de Recherche



Moteurs de Recherche

twitter @beyondsearch Search Advanced Search

Results for @beyondsearch 0.45 seconds

 [sitesdoneright](#): RT @otisg: Funny quote from [@beyondsearch](#): the reason there is proprietary software is to deliver a solution with "one throat to choke".
about 6 hours ago via HootSuite · [Reply](#) · [View Tweet](#)

 [otisg](#): Funny quote from [@beyondsearch](#): the reason there is proprietary software is to deliver a solution with "one throat to choke".
about 10 hours ago via HootSuite · [Reply](#) · [View Tweet](#)

 [dreasoning](#): RT @davednh: Does Facebook have a shot as a search/info leader? - RT @BeyondSearch: New blog post: Update on Facebook Questions <http://bit.ly/a4Zwh1> (expand)
1 day ago via TweetDeck · [Reply](#) · [View Tweet](#)

 [davednh](#): Does Facebook have a shot as a search/info leader? - RT @BeyondSearch: New blog post: Update on Facebook Questions <http://bit.ly/a4Zwh1> (expand)
1 day ago via TweetDeck · [Reply](#) · [View Tweet](#)

[Feed for this query](#)
[Tweet these results](#)

Show tweets written in:
Any Language

Trending topics:

- [Sufjan Stevens](#)
- [Harmoni SCTV](#)
- [#whybeinarelationshipif](#)
- [Rossi](#)
- [#happy18thbdydemil](#)
- [#mishabdayparade](#)
- [Inception](#)
- [Tiririca](#)
- [Gamal](#)
- [Dilma](#)

Moteur de Recherche

Exemple : Moteur de Recherche du Web

Trois étapes :

- ① Le Crawl : chercher les documents
- ② L'indexation : formatter les documents
- ③ La recherche : répondre à des requêtes (rapidement, et efficacement)

L'étape 3 est **critique** :

- Un utilisateur normal ne lit que le ou les premiers résultats
- Absolue nécessité d'un classement pertinent des réponses

Principes

Ordonnancement

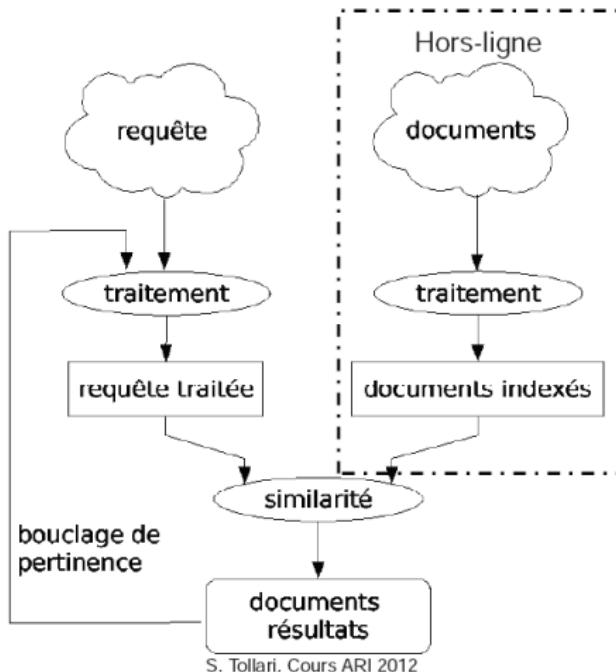
Pour chaque requête q et chaque document d , un moteur de recherche calcul un score de pertinence (RSV : Relevance Score Value) noté $RSV(q, d)$. Ce score permet de classer les résultats pour une requête q donnée par ordre décroissant de pertinence.

Contraintes :

- RSV doit pouvoir être calculé de manière rapide et distribuée
- RSV doit permettre d'obtenir de bons résultats

De très nombreux modèles ont été développés et évalués au cours des 30 dernières années (Domaine de recherche : **Information Retrieval (IR)**).

Principes



S. Tollari, Cours ARI 2012

Etapes

Moteur textuel

La construction d'un Moteur de Recherche Textuel s'effectue selon les étapes suivantes :

- Prétraitement des données textuelles (Preprocessing)
- Indexation globales des termes (*Document Frequency*)
- Indexation Inverse des documents (*Term Frequency*)
- Calcul du score (modèle *Cosine* ou modèle *Okapi*)
- **Evaluation du moteur de recherche**

Pretraitement

Normalisation (lemmatisation)

- Utilisation d'une forme canonique pour representer les variantes morphologiques d'un mot
- e.g. dynamic, dynamics, dynamically, ...seront représentés par un même mot, naviguer, naviguant, navireidem
- Augmente le rappel, peut diminuer la précision
- Techniques (exemples) :
 - systèmes itératifs a base de règles simples (e.g. pour l'anglais Porter stemming -largement employé) : on etabli une liste de suffixes et de prefixes qui sont élimines iterativement.
 - méthodes à base de dictionnaires mot - forme canonique. Intérêt : langue présentant une forte diversité lexicale (e.g. français)

Modèles de pondération : Loi de Zipf

Loi de Zipf

Zipf avait entrepris d'analyser une oeuvre monumentale de James Joyce, Ulysse, d'en compter les mots distincts, et de les présenter par ordre décroissant du nombre d'occurrences. La légende dit que :

- le mot le plus courant revenait 8 000 fois ;
- le dixième mot 800 fois ;
- le centième, 80 fois ;
- et le millième, 8 fois.

Modèles de pondération

Principe

Le score de pertinence est obtenue par une pondération des termes appartenant au document d et à la requête q :

Si une requête contient le terme T , un document a d'autant plus de chances d'y répondre qu'il contient ce terme : la fréquence du terme au sein du document (TF) est grande. Néanmoins, si le terme T est lui-même très fréquent au sein du corpus, c'est-à-dire qu'il est présent dans de nombreux documents (e.g. les articles définis - le, la, les), il est en fait peu discriminant.

Notations

TF : Term Frequency

Soit un document d et un mot w , on définit $tf(d, w)$ par :

$$tf(d, w) = \text{fréquence de } w \text{ dans } d$$

IDF : Inverse Document Frequency

Soit un mot w , on définit $idf(w)$ par :

$$idf(w) = \log \frac{|D|}{|\{d_j : w \in d_j\}|}$$

TF-IDF

TF-IDF

Le poids d'un mot dans un document est défini par :

$$tf - idf(w, d) = tf(w, d) * idf(w)$$

RSV

Le score de pertinence est :

$$RSV(d, q) = \sum_{w \in d, w \in q} tf - idf(w, d)$$

On n'appelle cela un **Modèle vectoriel** car $RSV(d, q)$ correspond à un produit scalaire entre un vecteur de document et un vecteur de requête (cf tableau)

Problématique

RSV

Le score de pertinence est :

$$RSV(d, q) = \sum_{w \in d, w \in q} tf - idf(w, d)$$

Problème :

Comment calculer cela efficacement, de manière distribué ?

Index inversé

Solution : Index inversé

- chaque terme de l'index est décrit par le numéro de référence de tous les documents qui contiennent ce terme et la position dans ce document du terme.
- Permet une accélération considérable de la recherche pour une requête. Cet index peut être ordonné en fonction décroissante de la fréquence des termes.
- Implémentation : différentes structures de données triées

Index inversé

- $T_0 = \text{"it is what it is"}$
- $T_1 = \text{"what is it"}$
- $T_2 = \text{"it is a banana"}$

"a": { $(2, 2)$ }

"banana": { $(2, 3)$ }

"is": { $(0, 1), (0, 4), (1, 1), (2, 1)$ }

"it": { $(0, 0), (0, 3), (1, 2), (2, 0)$ }

"what": { $(0, 2), (1, 0)$ }

Concrètement

Searvice Search

Le service SEARCH /ListMessage vise à permettre l'accès aux message postés sur le site.

Plusieurs modes de recherche :

- derniers message postés (page principale)
- à partir de la page profil d'un utilisateur
- Avec requêtes (mots clefs)
- Sans requêtes
- En incluant uniquement les amis
- ou pas.

Concrètement

Page principale

Le service renvoie les messages de tout le monde par ordre chronologique inverse (du plus récent au plus vieux)

Page profil

Le service renvoie les messages de l'utilisateur par ordre chronologique inverse (du plus récent au plus vieux)

Si une requête est spécifiée

L'ordre de retour des messages n'est plus l'ordre chronologique, mais l'ordre de pertinence : un score calculé entre la requête et chaque message

Concrètement

Entrées :

userID : id de l'utilisateur

key : clef de session (vide si pas authentifié)

query : la requête (vide si pas de requête)

friends : 0 ou 1 si on veut restreindre la recherche aux amis

Sorties :

Liste des messages dans le bon ordre (chronologique ou pertinence)

autres infos si besoin (query, contactOnly, author, date...)

ou

erreurs éventuelles

Concrètement

Objectif

Etapes pour notre site :

- Construction de l'index des *idf*
- Construction de l'index inversé
- Calcul de la fonction de score

Plusieurs Solutions

- Construction de l'index des *idf* = **Map-Reduce**
- Construction de l'index inversé = **Map-Reduce**
- Stockage des index :
 - Sous SQL ?
 - Sous MongoDB ?

Sous SQL

SQL

Avantages :

- le calcul du score de pertinence peut être effectué en une seule requête SQL
- L'optimisation est laissée à MySQL ⇒ très rapide

Inconvénients :

- Les index ne sont pas distribués...

Sous MongoDB

MongoDB

Inconvénients :

- le calcul du score doit être fait à travers plusieurs requêtes MongoDB
- ⇒ moins rapide

Avantages :

- Permet la gestion de très gros index

Conclusion

- Vous savez faire du Map-Reduce
- Vous savez ce qu'est un moteur de recherche
- Y a plus qu'à.....